

UNIVERSIDAD POLITÉCNICA SELESIANA  
SEDE QUITO

CARRERA:  
INGENIERÍA ELECTRÓNICA

Trabajo de titulación previo a la obtención del título de:  
INGENIERO ELECTRÓNICO

TEMA:  
CONTROL DE UN ROBOT HUMANOIDE PARA LA CATEGORÍA DE  
PELEA A TRAVÉS DE LOS MOVIMIENTOS ARTICULARES DEL  
OPERADOR

AUTOR:  
EDWIN GERMÁN SALVADOR MAYORGA

TUTORA:  
CARMEN JOHANNA CELI SÁNCHEZ

Quito, septiembre del 2019

## **CESIÓN DE DERECHOS DE AUTOR**

Yo Edwin Germán Salvador Mayorga, con documento de identificación N° 2000069498, manifiesto mi voluntad y cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del trabajo de titulación intitulado: “CONTROL DE UN ROBOT HUMANOIDE PARA LA CATEGORÍA DE PELEA A TRAVÉS DE LOS MOVIMIENTOS ARTICULARES DEL OPERADOR”, mismo que ha sido desarrollado para optar por el título de Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



Edwin Germán Salvador Mayorga

Cédula: 2000069498

Fecha: Quito, septiembre del 2019

## **DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR**

Yo declaro que bajo mi dirección y asesoría fue desarrollado el Proyecto Técnico, “CONTROL DE UN ROBOT HUMANOIDE PARA LA CATEGORÍA DE PELEA A TRAVÉS DE LOS MOVIMIENTOS ARTICULARES DEL OPERADOR” realizado por Edwin Germán Salvador Mayorga, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerados como trabajo final de titulación.

Quito, septiembre del 2019

A handwritten signature in blue ink, appearing to read 'Carmen Johanna Celi Sánchez Mgtr.', written over a horizontal line.

Ing. Carmen Johanna Celi Sánchez Mgtr.

CI: 1717437808

## **AGRADECIMIENTO**

Quiero agradecer a:

Dios por guiar mi camino y darme la oportunidad de llegar hasta donde he llegado y por darme salud.

A mi Directora de Tesis la Mgtr. Jhoanna Celi que me ha brindado un gran apoyo para alcanzar esta meta. A mi Director de Carrera el PhD. Ramón Pérez por todo el apoyo brindado a lo largo de mi carrera.

A mi familia y amigos por apoyarme y darme ánimos para continuar y alcanzar mis metas, por estar siempre pendientes de mí y brindarme todo su cariño.

A todos muchas gracias.

## **DEDICATORIA**

El presente trabajo se lo dedico a mi familia. A mis padres Germán Salvador y Mónica Mayorga por todo su amor, su ayuda, comprensión y apoyo a lo largo de mi vida, por siempre estar conmigo, por todos sus consejos y enseñanzas por nunca dejarme solo, por enseñarme a levantarme, por ser mi guía, por ser la luz en mi camino, por ser las personas con las que siempre puedo contar. A mi hermana Julia Salvador por su apoyo y comprensión, por estar a mi lado, por darme ánimos cuando más lo necesito, por cuidarme, por ser una gran hermana y amiga con el que puedo contar toda la vida. A mis abuelitos Carlos Mayorga y Juana Mayorga, que me enseñaron a luchar, a seguir adelante siendo ellos un gran ejemplo de superación, de persistencia, aunque ya no están a mi lado; sus enseñanzas, consejos, apoyo y todo su amor siempre estarán conmigo, y siempre vivirán en mi corazón. A todos mis maestros que por sus enseñanzas he logrado alcanzar el sitio en que me encuentro. A todos ellos muchas gracias.

## ÍNDICE DE CONTENIDOS

INTRODUCCIÓN .....	xii
<b>CAPÍTULO 1.....</b>	<b>1</b>
<b>ANTECEDENTES .....</b>	<b>1</b>
1.1 Planteamiento del Problema .....	1
1.2 Justificación del proyecto .....	1
1.3 Objetivos .....	1
1.3.1 Objetivo General: .....	1
1.3.2 Objetivos Específicos:.....	2
<b>CAPÍTULO 2.....</b>	<b>3</b>
<b>FUNDAMENTACIÓN TEÓRICA.....</b>	<b>3</b>
2.1 Sistemas Embebidos .....	3
2.1.1 Selección de Sistema Embebido .....	4
2.1.2 Arduino DUE .....	5
2.2 Módulo Bluetooth HC-06.....	7
2.3 Módulo Bluetooth HC-05.....	7
2.4 Controlador Power Rythm 32.....	8
2.5 Servos Digitales.....	9
2.6 Robot AI.frame Apollo.....	9
2.7 Sensores Flex.....	9
2.8 Sensores MMA 7361 .....	11
2.9 Multiplexor CJMCU-4051 .....	13
2.10 Batería de 7.4 voltios .....	15
2.11 Cinemática Directa .....	16
2.12 Robot humanoide .....	16
2.13 Articulaciones .....	17
2.13.1 El hombro.....	17
2.13.2 El codo .....	18

2.14	La cadera.....	19
<b>CAPÍTULO 3.....</b>		<b>20</b>
<b>DISEÑO E IMPLEMENTACIÓN .....</b>		<b>20</b>
3.1	Descripción del funcionamiento.....	20
3.2	Cinemática directa .....	21
3.3	Arquitectura.....	22
3.4	Acondicionamiento de sensores .....	25
3.4.1	Codo Izquierdo.....	27
3.4.2	Codo Derecho.....	27
3.4.3	Pierna Izquierda .....	27
3.4.4	Pierna Derecha .....	28
3.4.5	Cinemática directa del Brazo .....	28
3.4.6	Cinemática directa de la Pierna.....	32
3.5	Diagrama de flujo .....	34
3.6	Diagrama de bloques .....	36
3.7	Código fuente .....	37
3.8	Diseño del traje.....	40
<b>CAPÍTULO 4.....</b>		<b>45</b>
<b>PRUEBAS Y RESULTADOS .....</b>		<b>45</b>
4.1	Pruebas .....	45
4.2	Resultados .....	50
<b>CAPÍTULO 5.....</b>		<b>51</b>
<b>CONCLUSIONES.....</b>		<b>51</b>
<b>CAPÍTULO 6.....</b>		<b>53</b>
<b>RECOMENDACIONES.....</b>		<b>53</b>
<b>REFERENCIAS .....</b>		<b>54</b>

## ÍNDICE DE FIGURAS

Figura 2.1 Diagrama de asignación de pines .....	5
Figura 2.2 Partes Arduino DUE .....	6
Figura 2.3 Módulo Bluetooth HC-06 .....	7
Figura 2.4 Módulo Bluetooth HC-05 .....	8
Figura 2.5 Power Rythm 32 .....	8
Figura 2.6 Servo Digital .....	9
Figura 2.7 Sensor Flex .....	10
Figura 2.8 Sensor Flex .....	11
Figura 2.9 Sensor MMA 7361 .....	12
Figura 2.10 MMA 7361 .....	13
Figura 2.11 Diagrama funcional .....	14
Figura 2.12 Multiplexor .....	15
Figura 2.13 Robot Humanoide .....	17
Figura 2.14 Abducción, aducción y antepulsión .....	18
Figura 2.15 Codo .....	18
Figura 2.16 Cadera .....	19
Figura 3.1 Diagrama de flujo para el funcionamiento del robot .....	20
Figura 3.2 Ubicación de actuadores .....	21
Figura 3.3 Articulación .....	22
Figura 3.4 Arquitectura del Sistema .....	23
Figura 3.5 Traje Inferior .....	24
Figura 3.6 Curva eje X .....	25
Figura 3.7 Curva eje Y .....	25
Figura 3.8 Posiciones de motores .....	26
Figura 3.9 Cinemática del brazo .....	29
Figura 3.10 Posición inicial del brazo .....	29
Figura 3.11 Rotación del plano del brazo .....	29
Figura 3.12 Posición final del plano del brazo .....	30
Figura 3.13 Cinemática de la pierna .....	32
Figura 3.14 Posición inicial de la pierna .....	32
Figura 3.15 Posición final de la pierna .....	33
Figura 3.16 Flujo del programa .....	35
Figura 3.17 Diagrama de bloques .....	36



Figura 3.18 Salidas digitales .....	37
Figura 3.19 Lectura análoga.....	37
Figura 3.20 Control por histéresis .....	38
Figura 3.21 Condición.....	38
Figura 3.22 Código de envío .....	38
Figura 3.23 Final de la secuencia de comandos .....	39
Figura 3.24 Condiciones para levantar al robot .....	39
Figura 3.25 Interrupciones y cambio de estados .....	39
Figura 3.26 Caja contenedora .....	40
Figura 3.27 Traje brazo derecho .....	40
Figura 3.28 Hombro derecho .....	41
Figura 3.29 Codo derecho .....	41
Figura 3.30 Control de caídas .....	42
Figura 3.31 Hombro y codo izquierdo .....	42
Figura 3.32 Musleras.....	43
Figura 3.33 Traje completo .....	43
Figura 3.34 Arnés y hombrera con espaldar .....	44
Figura 4.1 Golpes Rectos .....	45
Figura 4.2 Movimiento combinado.....	46
Figura 4.3 Movimiento de abducción .....	46
Figura 4.4 Movimientos de Antepulsión.....	47
Figura 4.5 Movimiento lateral de piernas .....	48
Figura 4.6 Movimientos frontales de piernas.....	49
Figura 4.7 Golpe doble.....	49

## ÍNDICE DE TABLAS

Tabla 2.1 Tabla comparativa de sistemas embebidos Arduino.....	4
Tabla 2.2 Tabla de Verdad .....	14
Tabla 3.1 Valores análogos del sensor .....	24
Tabla 3.2 Parámetros Denavit-Hartenberg (brazo) .....	30
Tabla 3.3 Parámetros Denavit-Hartenberg (pierna) .....	33
Tabla 4.1 Tiempo de respuesta.....	50

## **RESUMEN**

Los operadores de robots de pelea necesitan poseer grandes habilidades en el manejo de dispositivos inteligentes o joystick para controlar los movimientos de los robots, además debían poseer gran experiencia para evitar errores y un mal desempeño del robot.

En el presente proyecto se diseñó un traje que será utilizado por un operador para controlar un robot bípedo y solucionar el inconveniente de los operadores con pocas habilidades en el manejo de controles, permitiendo al usuario tener una mayor libertad de movimientos y autonomía.

El traje diseñado posee articulaciones con sensores flex y giroscópicos calibrados tal manera que receptan la información de los movimientos realizados, el comportamiento de dichos sensores es lineal en el área de trabajo requerida, los datos obtenidos son procesados y transmitidos mediante el sistema embebido hacia el robot “AI.frame Apollo” utilizando comunicación inalámbrica mediante los módulos Bluetooth HC 05 ubicados en el sistema embebido y en el robot respectivamente, dichos módulos fueron configurados tal manera que al encender el robot y el sistema embebido estos se conecten automáticamente, el robot recibe la información dando como resultado el movimiento de la articulación deseada.

Los resultados obtenidos permitieron evidenciar la ventaja del traje ante la utilización de dispositivos controladores, esto se debe a que no se requiere experiencia ya que los movimientos son naturales. Los requerimientos para manipular el robot son: el operador debe tener cierto conocimiento en combate cuerpo a cuerpo, una breve introducción acerca de la utilización del robot y colocar correctamente el traje.

## **ABSTRACT**

The operators of fighting robots need to have great skills in the management of smart devices or joystick to control the movements of the robots, in addition they had to have great experience to avoid errors and a bad performance of the robot.

In this project, a suit was designed that will be used by an operator to control a bipedal robot and solve the inconvenience of operators with few control management skills, allowing the user to have greater freedom of movement and autonomy.

The designed suit has joints with flex and gyroscopic sensors calibrated in such a way that they receive the information of the movements made, the behavior of said sensors is linear in the required work area, the data obtained is processed and transmitted by the system embedded to the robot “AI.frame Apollo” using wireless communication through the Bluetooth HC 05 modules located in the embedded system and in the robot respectively, these modules were configured such that when the robot is turned on and the embedded system these are automatically connected, the robot receives the information resulting in the movement of the desired joint.

The results obtained allowed to demonstrate the advantage of the suit before the use of controlling devices, this is because no experience is required since the movements are natural. The requirements to manipulate the robot are: the operator must have some knowledge in close combat, a brief introduction about the use of the robot and correctly place the suit.

## INTRODUCCIÓN

El presente trabajo tiene como objetivo principal controlar un robot humanoide de pelea utilizando un sistema embebido con cálculos de cinemática directa, el robot imitará los movimientos de un operador, el operador utilizará un traje en donde se encontrarán colocados los sensores. El traje estará conectado a un sistema embebido el cual receptará la información enviada por el robot mediante Bluetooth.

En el capítulo 1 se describe la problemática de los usuarios de robots de pelea al no poseer grandes habilidades en el manejo de dispositivos inteligentes o joystick para controlar y manipular a los robots, también se describe los objetivos y justificación para la realización del presente trabajo.

En el capítulo 2 se detalla la fundamentación teórica de los implementos y equipos utilizados, los cuales servirán como base para el desarrollo del capítulo 3 en donde se explica la implementación de dichos equipos, así como los cálculos de cinemática directa y el acondicionamiento de los sensores para la transmisión de los datos desde el operador hacia el robot, dicha transmisión de datos se realizará mediante comunicación inalámbrica, específicamente comunicación Bluetooth.

Para finalizar se realizará las pruebas pertinentes para comprobar que el robot es capaz de imitar los movimientos realizados por el operador.

# **CAPÍTULO 1**

## **ANTECEDENTES**

### **1.1 Planteamiento del Problema**

Cuando los estudiantes de robótica o los aficionados a las peleas de robots humanoides que manipulan sus robots mediante dispositivos inteligentes o joystick, se encuentran con la limitación de los movimientos preestablecidos en las aplicaciones o controles, las mismas que deben encontrarse actualizadas para un correcto funcionamiento y la persona que controle el robot debe tener una gran habilidad y experiencia en el manejo de robots para realizar una correcta utilización de los comandos de control y evitar cometer errores por presionar un botón erróneamente. (Torrejón, 2011)

### **1.2 Justificación del proyecto**

El presente proyecto será desarrollado para facilitar el manejo de un robot bípedo para un operador que no posea grandes destrezas frente a un dispositivo inteligente o joystick; el robot está limitado a los movimientos predefinidos por el programador, si el usuario necesita nuevos movimientos, no podrá obtenerlos de forma inmediata y muchos menos en medio de una competencia. Por el contrario si se utiliza el proyecto propuesto, el usuario podrá crear nuevos movimientos según sea las circunstancias y experiencia en combate, si el usuario tiene experticia en artes marciales, lo puede manejar de mejor manera ya que puede implementar combinaciones o movimientos que utilizan en combates de la vida real, lo cual proporciona una ventaja dentro de una competencia. (Kopiev, 2017)

### **1.3 Objetivos**

#### **1.3.1 Objetivo General:**

Controlar un robot humanoide a través de los movimientos articulares del operador utilizando un sistema embebido para la categoría de pelea de humanoides realizando cinemática directa.

### **1.3.2 Objetivos Específicos:**

- Investigar el funcionamiento de los sensores que serán los encargados de adquirir la información de los movimientos realizados por el operador para el control del robot
- Investigar acerca de un sistema embebido para el procesamiento de la información adquirida por los sensores.
- Diseñar el traje que controle un robot humanoide para la imitación de los movimientos del operador
- Comunicar el sistema embebido con el traje para el procesamiento de los movimientos y enviarlos al robot vía inalámbrica
- Desarrollar la cinemática directa para el control de los movimientos del robot humanoide
- Realizar las pruebas del funcionamiento del traje y robot en una competencia de pelea frente a un adversario real.

## CAPÍTULO 2

### FUNDAMENTACIÓN TEÓRICA

#### 2.1 Sistemas Embebidos

Los sistemas embebidos son equipos electrónicos que han sido diseñados y desarrollados para realizar funciones específicas en tiempo real.

Los sistemas embebidos cuentan con una placa base en la cual se puede encontrar la mayoría de sus componentes. Algunos ejemplos de sistemas embebidos pueden ser taxímetros, controles de acceso, la electrónica que controla una máquina expendedora, etc. (Departamento de Ingeniería Eléctrica, Electrónica y de Control, 2017)

Un sistema embebido tiene como característica principal un microcontrolador que es el cerebro de todo el sistema, este no es nada más que un microprocesador que tiene en su arquitectura entradas y salidas digitales o análogas; los sistemas embebidos se pueden programar con lenguajes como C, C++ o lenguajes interpretados como JAVA. (Departamento de Ingeniería Eléctrica, Electrónica y de Control, 2017)

Las características de los sistemas embebidos según la referencia son (Bricogeek, 2015):

- Bajo costo

Su bajo costo se debe a que su producción y consumo es masivo, poniendo la tecnología de estos sistemas al alcance de todos, permitiendo así un desarrollo continuo.

- Bajo consumo de energía.

El bajo consumo de energía es producido por poseer placas de tamaño reducido con componentes de tecnología SMD, otros factores que influye en el bajo consumo de energía es que no disipan grandes cantidades de calor y que su voltaje de operación es de 3.3v en la mayoría de los casos.

Todo sistema embebido necesita un lugar donde alojar el programa a ser ejecutado lo que da lugar a las memorias RAM y ROM (Sistemas Embebidos: Innovando hacia los Sistemas Inteligentes, 2016)



### 2.1.1 Selección de Sistema Embebido

En la Tabla 2.1 se realiza una comparación entre los sistemas embebidos Arduino: UNO, MEGA y DUE; la tabla comparativa permitirá elegir el sistema embebido idóneo para el procesamiento de la información. (Arduino, 2019)

Tabla 2.1 Tabla comparativa de sistemas embebidos Arduino

	Arduino UNO	Arduino MEGA	Arduino DUE
<b>Procesador</b>	Atmega 328	Atmega 2560	AT91SAM3X8E
<b>Reloj</b>	16 MHz	16 MHz	84 MHz
<b>Memoria Flash (kb)</b>	32	256	512
<b>EEPROM (kb)</b>	1	4	N/D
<b>SRAM (kb)</b>	2	8	96
<b>Nivel de voltaje</b>	5V	5V	3.3V
<b>Corriente máxima pin 3.3V</b>	50 mA	50 mA	800 mA
<b>Pines I/O digitales</b>	14	54	54
<b>PWM</b>	6	15	12
<b>Pines análogos</b>	6	16	12
<b>DAC</b>	0	0	2

Tabla comparativa entre Arduino: UNO, MEGA y DUE, Elaborado por: Edwin Salvador

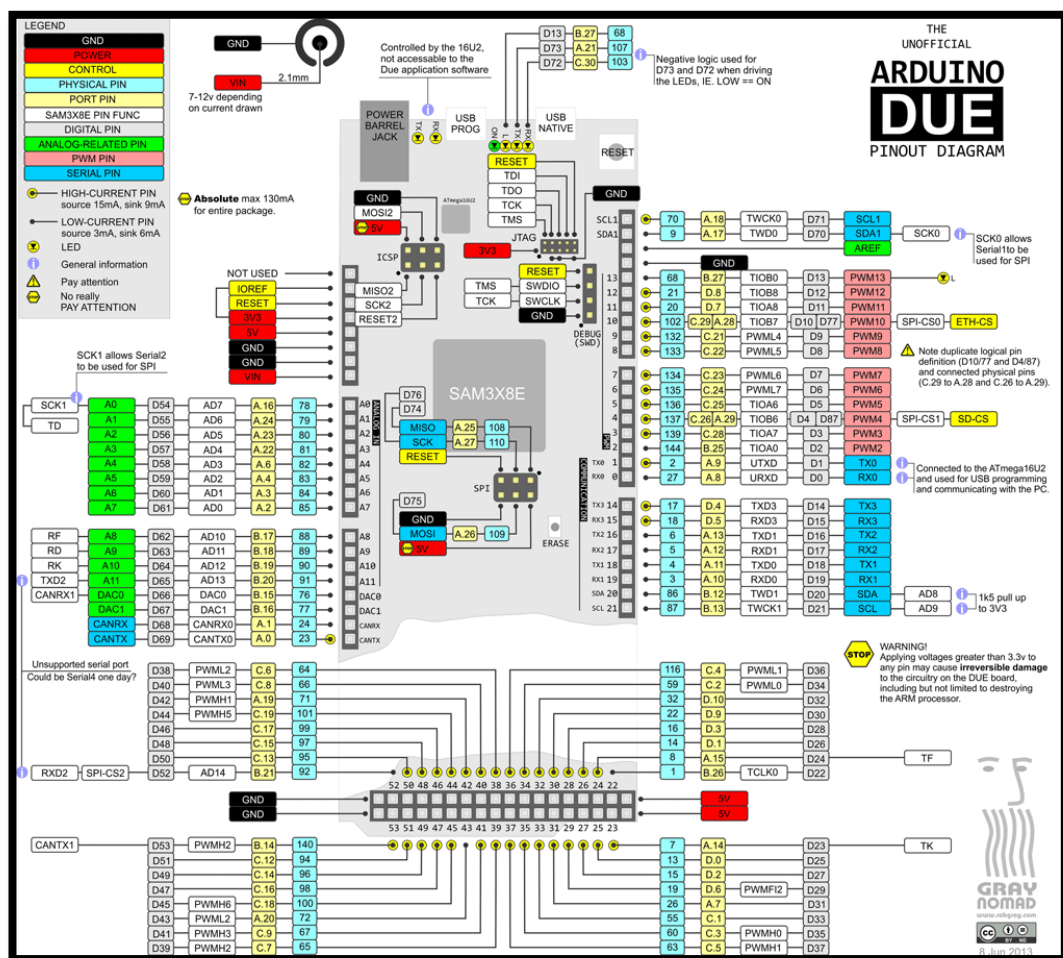
Según la información de la Tabla 2.1 se observa que el sistema embebido que cubre las necesidades del proyecto es el Arduino DUE, por sus características como son: la velocidad del reloj, debido a que su tiempo de procesamiento es mucho más corto por consecuencia su reacción es mucho más rápida; su voltaje de operación, ya que a menor voltaje trabajado la disipación de calor es mucho menor esto conlleva a un trabajo más eficiente energéticamente; su memoria flash, la capacidad de la memoria es suficiente para permitir almacenar el programa descargado sin ningún problema; su máxima corriente de salida en el pin de 3.3V, debido a que este pin alimentará a los sensores que se encuentran ubicados en el traje ya que su corriente máxima de salida es de 800 mA. (Arduino, 2019)

## 2.1.2 Arduino DUE

Arduino DUE es un microcontrolador basado en la CPU Atmel SAM3X8E ARM Cortex-M3, que se programa utilizando el software de Arduino (IDE), su velocidad de funcionamiento está basado en un reloj que proporciona una frecuencia de 84MHZ a 32bit. (SM, 2017)

Posee 54 pines de entradas o salidas digitales, 12 de aquellos pines se los puede utilizar como entradas analógicas, 12 como salidas PWM, 4 pines para comunicación seria (UARTs), 2 pines como salidas análogas (DAC) y 2 pines como puertos de can bus el cual nos sirve para hacer una conexión directa a las líneas de comunicación de los vehículos. En la Figura 2.1 se observa la distribución mencionada en el diagrama de Pines del Arduino DUE. (SM, 2017)

Figura 2.1 Diagrama de asignación de pines



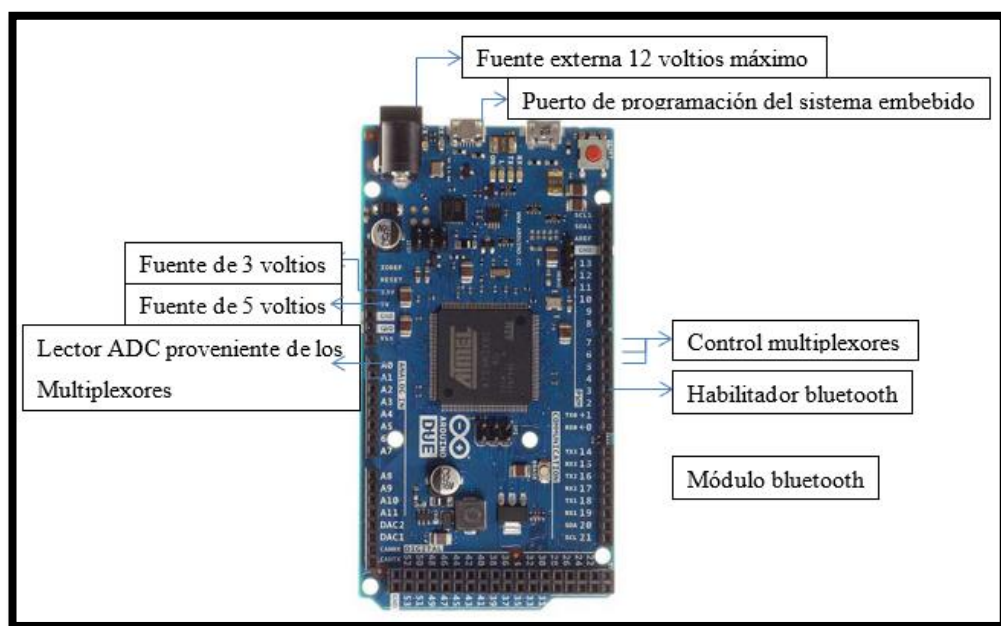
Arduino DUE, asignación de pines, Fuente:(Graynomad, 2013)

Se puede observar una distribución de pines, una característica relevante que es el voltaje máximo de entrada de hasta 12V, mientras en otras placas el voltaje de entrada máximo puede ser de 16V, la placa trabaja con 3.3V, pese a esto tiene un pin de alimentación que proporciona 5v para conectar sensores o actuadores. Posee un botón de borrado, ya que previa descarga de programa, su memoria debe ser borrada, esto se debe a que tiene un procesador ARM y no se puede programar directamente. (SM, 2017)

El sistema embebido Arduino DUE es ideal para proyectos que tienen gran cantidad de datos y su tiempo de respuesta debe ser corto como es el caso del presente proyecto que necesita adquirir información de los sensores a alta velocidad ya que por motivos de estar en un combate no puede tomar tiempos prolongados para la adquisición de la información. (SM, 2017)

Se eligió Arduino DUE debido a su arquitectura y a su capacidad de procesamiento, ya que cuenta con un microprocesador de 32 bits, el cual puede hacer multiplicaciones en un solo ciclo de reloj, y esto favorece al momento de realizar las operaciones de la cinemática directa. En la Figura 2.2 se puede observar las partes principales de un Arduino DUE. (SM, 2017)

Figura 2.2 Partes Arduino DUE

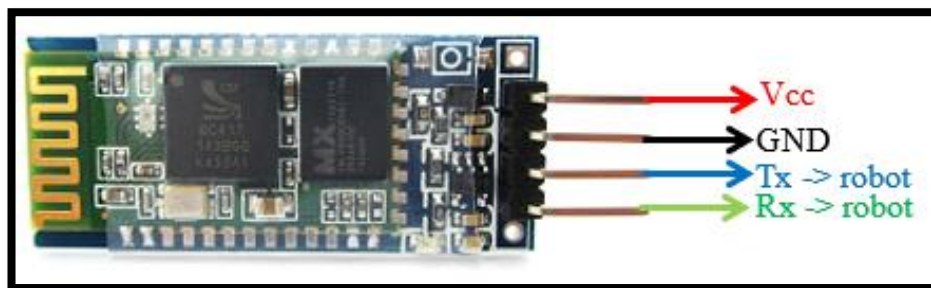


Partes principales de un Arduino DUE, Elaborado por: Edwin Salvador

## 2.2 Módulo Bluetooth HC-06

El módulo bluetooth que tiene el robot pre instalado, no es apto para comunicarse con cualquier otro dispositivo, únicamente se puede sincronizar con módulos bluetooth de la misma marca o en su defecto con dispositivos inteligentes, por ese motivo se cambia por un HC-06 el cual al momento de sincronizar con otro dispositivo lo hace de manera correcta y con la facilidad deseada. En la Figura 2.3 se observa los pines del módulo bluetooth HC-06. (Electronicos Caldas, S/F)

Figura 2.3 Módulo Bluetooth HC-06

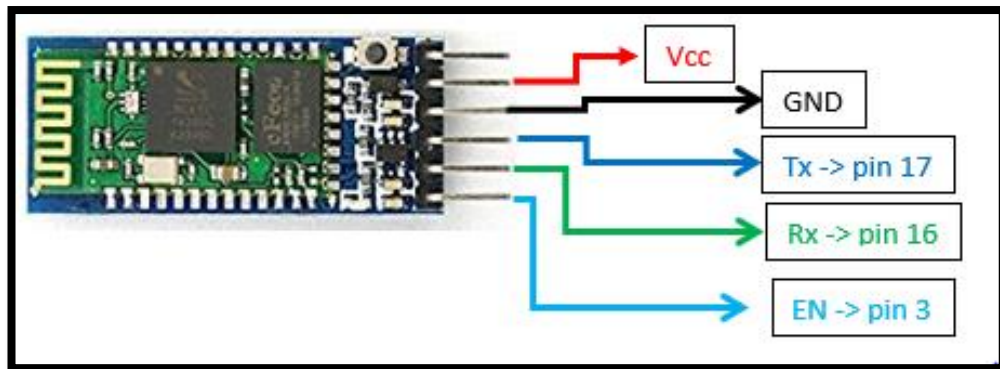


Pines del módulo Bluetooth HC-06, Elaborado por: Edwin Salvador

## 2.3 Módulo Bluetooth HC-05

Se elige la utilización del módulo HC-05 porque posee una facilidad para su conexión con otros módulos mediante comando “AT”, este módulo se lo utiliza para enviar la información desde el sistema embebido hacia el robot, el módulo HC-05 al momento de realizar el emparejamiento con otro dispositivo bluetooth es muy estable su conexión y la única forma de romper su conexión es cuando uno de los dos el emisor o el receptor tienen una caída de corriente o sufren una desconexión. En la Figura 2.4 se observa los pines del módulo bluetooth HC-05. (Components10, S/F)

Figura 2.4 Módulo Bluetooth HC-05

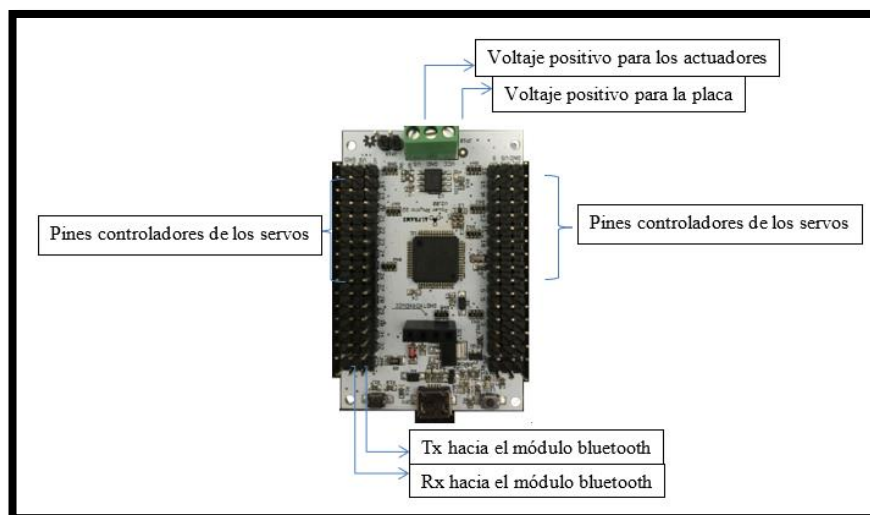


Pines del módulo Bluetooth HC-05, Elaborado por: Edwin Salvador

## 2.4 Controlador Power Rythm 32

Power Rythm 32 es un controlador de 32 servos que se encuentra incluido en el kit del robot AI.frame Apollo, y se decidió utilizar debido a su compatibilidad y las instrucciones que recibe son similares a la de otras controladoras, se utiliza comunicación serial esto es debido a que los módulos bluetooth utilizan este tipo de comunicación, a una velocidad de 115200bps. En la Figura 2.5 se observa las partes principales de un Power Rythm 32. (AI.Frame, 2016)

Figura 2.5 Power Rythm 32



Partes principales del Power Rythm 32, Elaborado por: Edwin Salvador

## 2.5 Servos Digitales

Los servos digitales son utilizados debido a la precisión que proporcionan, este tipo de servos no trabajan únicamente con el ancho de pulso (PWM) sino que también trabajan con una serie de pulsos, los cuales le dan la posición al servo, cuentan con un microcontrolador integrado para que pueda interpretar toda esta información. Los servos que vienen incluidos con el robot son digitales, además son los ideales para su funcionamiento, por ese motivo no han sido reemplazados. En la Figura 2.6 se puede observar un servo digital con la identificación de sus cables. (Grupo de Robots Argentina, 2018)

Figura 2.6 Servo Digital



Servo Digital, Elaborado por: Edwin Salvador

## 2.6 Robot AI.frame Apollo

El robot Apollo de la marca AI.FRAME posee código abierto, lo que proporciona cierta flexibilidad para adaptarse con cualquier otra tecnología de código abierto tal como el Arduino, además su estructura es de plástico PBS lo que brindando flexibilidad y resistencia a las caídas, esto es ideal ya que al ser un robot de pelea este está sometido a mucho maltrato físico en caso de competencias o incluso al momento entrenar los movimientos. (AI.Frame, 2016)

## 2.7 Sensores Flex

Un sensor Flex cambia su resistencia de acuerdo al grado con el que se dobla; transformando la curvatura en distintos valores de resistencia, trabajan con divisores de tensión para cuando cambie el valor de su resistencia, el valor de la tensión se vea alterado para de esa manera poder realizar una conversión ADC con el microcontrolador. (Rambal, 2017). En la Figura 2.7 se observa un Sensor Flex

Figura 2.7 Sensor Flex

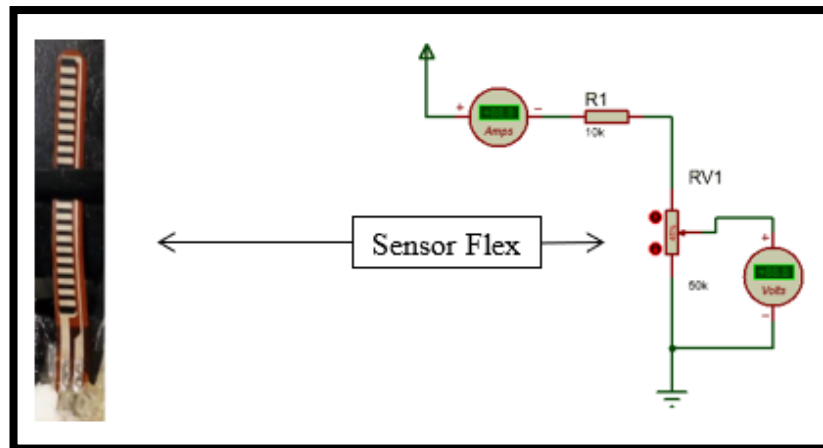


Sensor Flex – apariencia, Elaborado por: Edwin Salvador

Los sensores Flex tienen composición de carbono, a mayor cantidad de carbono menor la resistencia, la resistencia es directamente proporcional a la curvatura, se lo relaciona también con un potenciómetro y en ciertas ocasiones se lo llaman potenciómetro flexible, su tiempo de respuesta es muy bajo por ese motivo es utilizado en mediciones de impacto a alta velocidad, para determinar cuántos grados y con qué aceleración se deformó un objeto. (Sensor Products & INC, 2019)

Los sensores Flex son una opción válida para el proyecto ya que su tiempo de respuesta es bajo, esto se da a notar al momento de realizar los movimientos, el sensor Flex no tarda en regresar a su posición inicial, está compuesto de una tira flexible de carbono, lo que indica que mientras más carbono menos resistencia, el sensor Flex varía su resistencia entre  $10K\Omega$  a  $50k\Omega$  aproximadamente según el ángulo de curvatura física a la cual este sometido, su modo de funcionamiento es con un divisor de tensión como se muestra en la Figura 2.8. (Sensor Products & INC, 2019)

Figura 2.8 Sensor Flex



Divisor de tensión de un sensor Flex, Elaborado por: Edwin Salvador

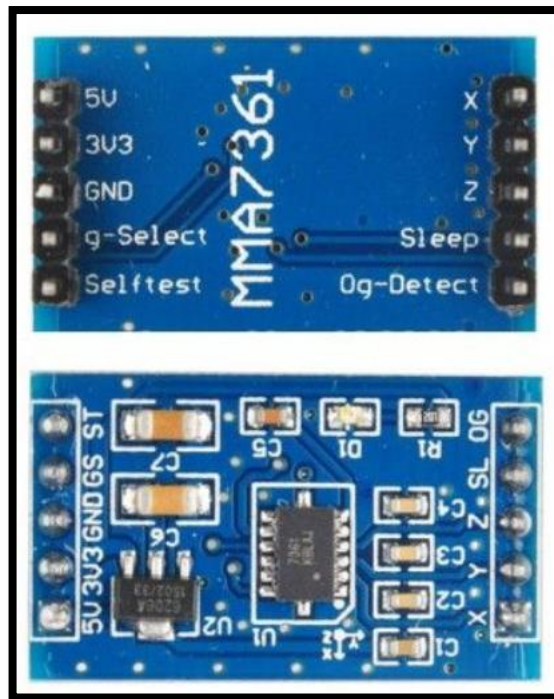
Según las especificaciones del fabricante, a mayor grado de deformación, su resistencia incrementa hasta llegar a  $180^\circ$  que es la curva máxima que puede alcanzar el sensor, cuando la curvatura del sensor llega a los  $180^\circ$ , la resistencia se eleva a  $50k\Omega$ . (Sensor Products & INC, 2019)

## 2.8 Sensores MMA 7361

El sensor MMA 7361, es un acelerómetro analógico de 3 ejes (x, y, z) según las medidas del acelerómetro como se puede observar en la Figura 2.9, se puede medir la aceleración o la inclinación de una superficie con respecto al centro de la tierra. (MIT MOVILTRONICS Ltda)



Figura 2.9 Sensor MMA 7361



Sensor MMA 7361 vista anterior y posterior, Fuente: (MIT MOVILTRONICS Ltda)

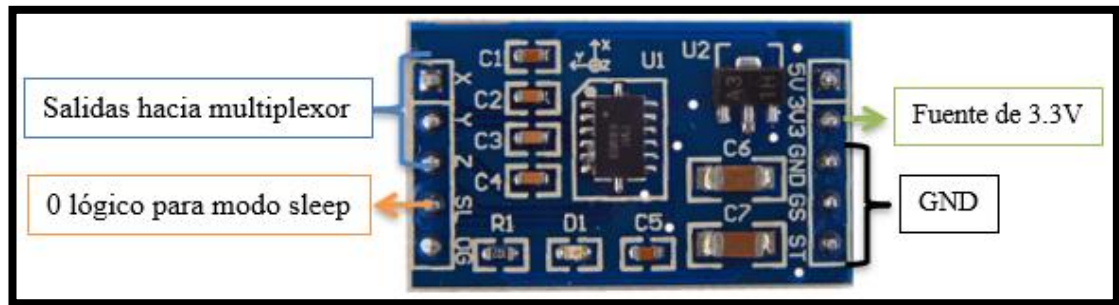
Entre sus características, el sensor tiene un filtro pasa-bajos de 1 polo, el cual da un acondicionamiento de la señal, además tiene compensación de temperatura para que esta no le afecte al momento de su funcionamiento y un detector de 0g (cero gravedad) para cuando se encuentre en caída libre, el acelerómetro MMA7361 es ideal para proyectos de robótica, controles remotos sensibles al movimiento o incluso interfaces gestuales, con las cuales se han desarrollado varios proyectos como son el manejo de un dron únicamente con el movimiento de la mano. (Haro, 2015)

Se selecciona la placa MMA 7361 por poseer salidas análogas, con un consumo energético bajo de alrededor de 400uA y una sensibilidad alta de 800mV/g a 1.5G; su voltaje de funcionamiento es de 3.3V, pero posee un pin de 5V, este pin está conectado a un regulador que lo reduce a 3.3V. Además, cuenta con un filtro pasa bajos para de esa manera filtrar la información que se proporciona y no tener en la salida mucha variación, sino que sea lo más estable posible. (Haro, 2015)

Esta placa es de gran utilidad ya que para comunicarlo con el sistema embebido no es necesario establecer protocolos de comunicación como por ejemplo el I2C que al momento de utilizar dos placas iguales, surge el inconveniente de las direcciones, esto

sucede si son dos placas de las mismas ya que las direcciones también son las mismas, pero si son análogas basta con leer de forma adecuada sus salidas e interpretar de manera correcta los datos obtenidos, para de esa manera poder generar conflictos en su utilización. En la Figura 2.10 se observa la distribución de los pines para la utilización de la placa MMA 7361 (Haro, 2015)

Figura 2.10 MMA 7361



Distribución de pines, Elaborado por: Edwin Salvador

Esta placa se la utiliza para detectar los movimientos de las piernas específicamente, el hueso femoral tiene dos grados de libertad, se escogió utilizar la placa MMA7361, la solución se presenta cuando se utiliza una placa ya que se la puede sujetar de mejor manera en una base plástica plana y así poder leer los datos de los ángulos de inclinación que se generan al momento de que el piloto o el usuario del traje se mueva. (Haro, 2015)

## 2.9 Multiplexor CJMCU-4051

El multiplexor CJMCU-4051 es un multiplexor de 8 canales a 1, las entradas a ser multiplexadas son análogas, con un consumo relativamente bajo por lo cual es una tarjeta ideal para ser utilizada con microcontroladores. Cuando una entrada no está activada, la misma se pone automáticamente en alta impedancia de esa manera no permite el paso de corriente y evita errores al momento de la lectura con un microcontrolador. (UNISONIC TECHNOLOGIES CO., LTD, 2005)

Éstos dispositivos pueden funcionar con 3.3V lo que es ideal para el funcionamiento con un Arduino DUE el cual funciona con 3.3V, si se aplica un voltaje TTL este se quema, su velocidad de multiplexación es muy elevada y esto se debe a que trabaja con tecnología CMOS, además su corriente de funcionamiento es de 10mA y su corriente de swicheo es de 25mA. (Texas Instruments Incorporated, 2017)

En la tabla 2.2 se encuentra detallado el funcionamiento lógico del multiplexor según la variación de sus entradas digitales mediante una tabla de verdad.

Tabla 2.2 Tabla de Verdad

Entradas de control			A	ENTRADAS
Inhibido	C	B		ANÁLOGAS
0	0	0	0	X0
0	0	0	1	X1
0	0	1	0	X2
0	0	1	1	X3
0	1	0	0	X4
0	1	0	1	X5
0	1	1	0	X6
0	1	1	1	X7
1	X	X	X	NADA

Tabla de verdad del multiplexor, Elaborado por: Edwin Salvador

En la Figura 2.11 se observa el diagrama funcional que utiliza el multiplexor, éste diagrama viene dado por el fabricante.

Figura 2.11 Diagrama funcional

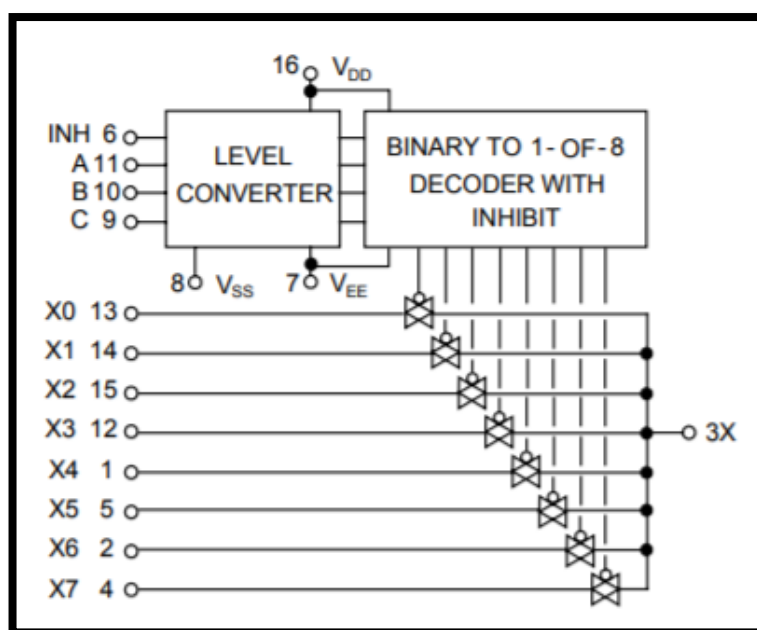
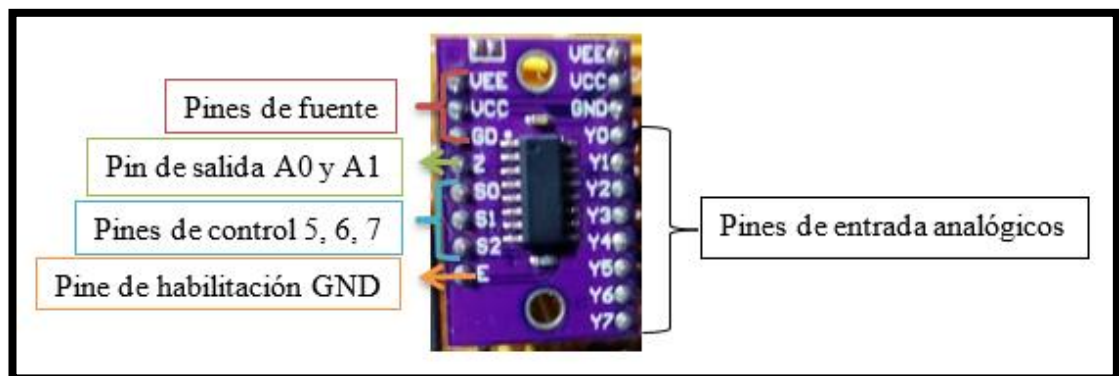


Diagrama funcional del multiplexor, Fuente: (UNISONIC TECHNOLOGIES CO., LTD, 2005)

El multiplexor 4051 proporciona 8 entradas analógicas, una salida y pines de control, además de un pin de habilitación (como se observa en la Figura 2.12), para utilizar los pines de selección únicamente se debe controlar tres bits, una de sus características es que puede trabajar con 3.3 voltios, que es el voltaje que suministra el sistema embebido.

Figura 2.12 Multiplexor



Descripción de pines del multiplexor, Elaborado por: Edwin Salvador

Su utilización en el proyecto es muy conveniente ya que es un dispositivo de alta velocidad debido a que posee una arquitectura basada en CMOS y además es compatible con TTL, ya que es un dispositivo de alta velocidad, los pines de habilitación pueden interactuar en tiempo real con el sistema embebido. (Texas Instruments Incorporated, 2017)

El multiplexor 4051 se activa directamente con el sistema embebido por su voltaje de 3.3v. La corriente de entrada del multiplexor es de 10 mA, como se utilizan dos multiplexores iguales, su corriente total es de 20 mA lo cual es una corriente manejable para el sistema embebido, ya que el sistema embebido nos proporciona 800mA como máximo. (Texas Instruments Incorporated, 2017)

## 2.10 Batería de 7.4 voltios

Las baterías se utilizan para suministrar energía, son celdas electroquímicas y son de gran utilidad para los equipos que requieran prolongados tiempos de uso o tengan un alto consumo eléctrico en donde las pilas de electrolitos no puedan suministrar la

energía necesaria. Los tipos de pilas secas más comunes son la de zinc-carbono y la pila alcalina; por el hecho de ser recargables producen menos desechos, por lo cual el impacto ambiental es menor. (Curiosoando.com, 2014)

### **2.11 Cinemática Directa**

La cinemática directa permite encontrar la posición y orientación que tendrá un robot al final de su movimiento, sin considerar la fuerza que lo produce; la posición final que tendrá el robot va de acuerdo al sistema de coordenadas iniciales utilizadas, esto debido a que el robot está formado por eslabones y articulaciones. (Ramírez Benavides)

Es necesario utilizar como herramienta fundamental las matrices de rotación y traslación, las que unidas general la matriz de transformación de coordenadas homogéneas, conocida como matriz de “Denavit-Hartenberg”. (Universidad de Santiago de Chile, 2018). Su proceso de análisis se encuentra detallado en el capítulo 3.

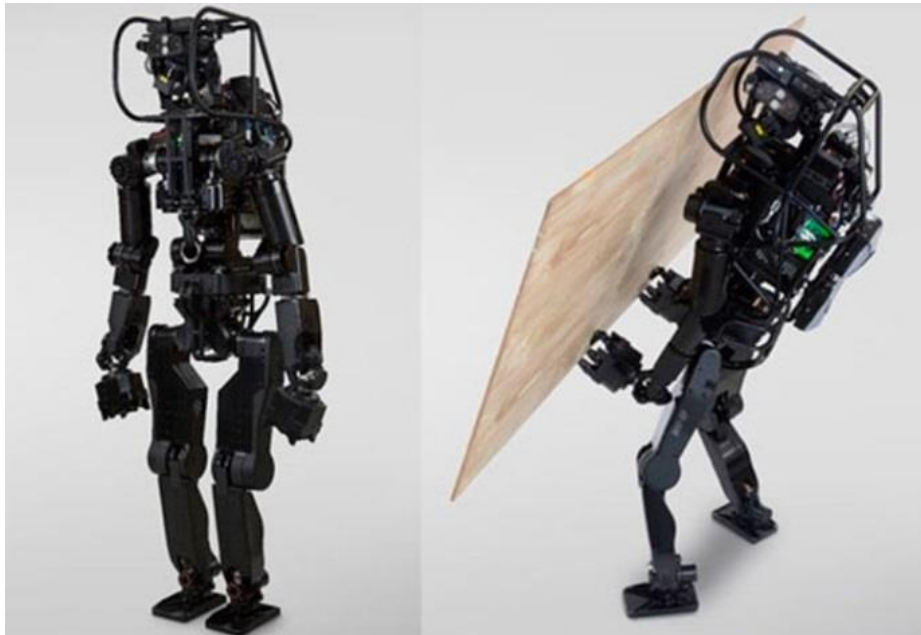
### **2.12 Robot humanoide**

Un robot humanoide es una máquina que simula la morfología y movilidad de una persona, son empleados en todo tipo de experimentos debido a que son máquinas programables, este tipo robots se los suele utilizar en combates, algunos robots son de código abierto los cuales se los programa con un software desde el Pc para realizar los movimientos son utilizados en diversas categorías o para desarrollar el pensamiento creativo de los niños. (RobotHumanoide.Top, 2018)

Los robots humanoides son el futuro tecnológico, pero teniendo los conocimientos necesarios e interactuando de una manera correcta con las máquinas, pueden llegar a ser un soporte inteligente para la humanidad ya que se puede llegar a desarrollar una inteligencia artificial para que la convivencia con estas máquinas sea el ideal, e incluso podrían llegar a reemplazar a los humanos en los trabajos de riesgo o que se necesite mucha fuerza para realizarlos. (Computer Hoy, 2018)

En la Figura 2.13 se puede observar a un robot humanoide realizando un trabajo de fuerza para mover una tabla.

Figura 2.13 Robot Humanoide



Robot humanoide realizando trabajo de fuerza, Fuente: (Computer Hoy, 2018)

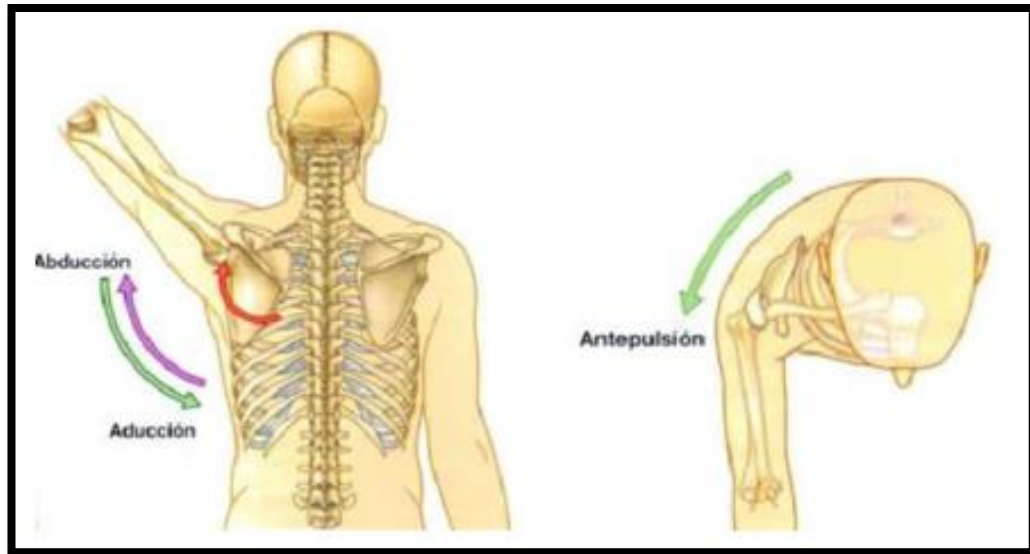
## 2.13 Articulaciones

### 2.13.1 El hombro

Los hombros tienen tres grados de libertad (como se puede observar en la Figura 2.14) para el movimiento, los mismos que serán captados por dos sensores Flex, los sensores captarán la siguiente información (Castillo Rodríguez, 2012):

- Aducción y Abducción: el sensor captará los movimientos en los cuales el húmero tiende a bajar (aducción) o subir (abducción).
- Antepulsión: el sensor capta los movimientos en el cual el humero está extendido o en forma paralela al piso y su movimiento es de adelante hacia atrás o viceversa.

Figura 2.14 Abducción, aducción y antepulsión



Grados de libertad del hombro, Fuente: (Castillo Rodríguez, 2012)

### 2.13.2 El codo

El codo humano y el codo del robot son articulaciones que poseen un grado de libertad de movimiento, por lo que se optó por ubicar un sensor Flex en el traje para que este sensor sea el encargado de recibir la información del movimiento realizado por el codo del operador y los transmita al sistema embebido para de esa manera ser enviado al robot y pueda imitar la acción, la mecánica del codo se asemeja a una bisagra como se observa en la Figura 2.15. (Acero Jáureguil, 2013)

Figura 2.15 Codo



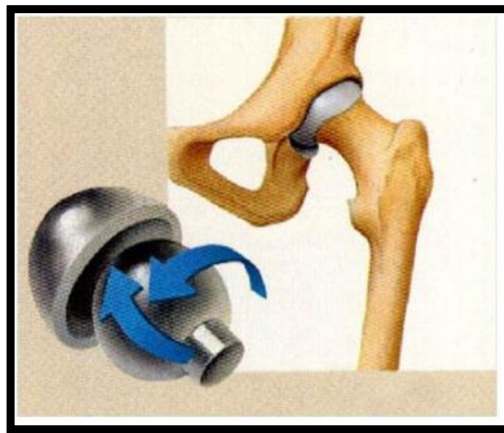
Grado de libertad del codo, Fuente: (Acero Jáureguil, 2013)

## 2.14 La cadera

Es una articulación con tres grados de libertad, dos rotaciones y una traslación (Figura 2.16), los mismos que serán captados por dos placas MMA 7361 que estarán ubicadas en el traje inferior, una placa en el lado derecho y otra placa en lado izquierdo, captando la siguiente información en cada lado (Acero Jáureguil, 2013):

- Rotación: los movimientos de la cadera de izquierda a derecha.
- Traslación: los movimientos de la cadera de adelante hacia atrás.

Figura 2.16 Cadera



Grados de libertad de la cadera, Fuente: (Acero Jáureguil, 2013)

Los movimientos se detectarán con el sensor giroscópico (placa MMA 7361), porque se puede medir el ángulo de inclinación, en este caso se medirá los ángulos en el eje “X” (ROLL) y en el eje “Y” (PITCH); al eje “Z” (YAW) no se lo tomará en cuenta ya que la cadera del robot no tiene posibilidad alguna de rotar en ese eje, esto se debe a que su arquitectura física no ha sido diseñada para eso, en el caso del operador si puede hacer los giros de su cadera pero el traje no censará dichos movimientos y no los tomara en cuenta.



## CAPÍTULO 3

### DISEÑO E IMPLEMENTACIÓN

#### 3.1 Descripción del funcionamiento

Para realizar el proceso del control de robot humanoide se debe describir el proceso completo y las etapas necesarias para el funcionamiento del robot.

Figura 3.1 Diagrama de flujo para el funcionamiento del robot

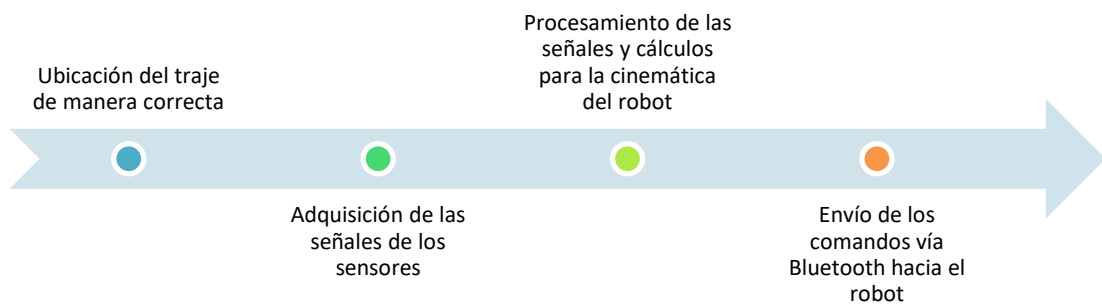
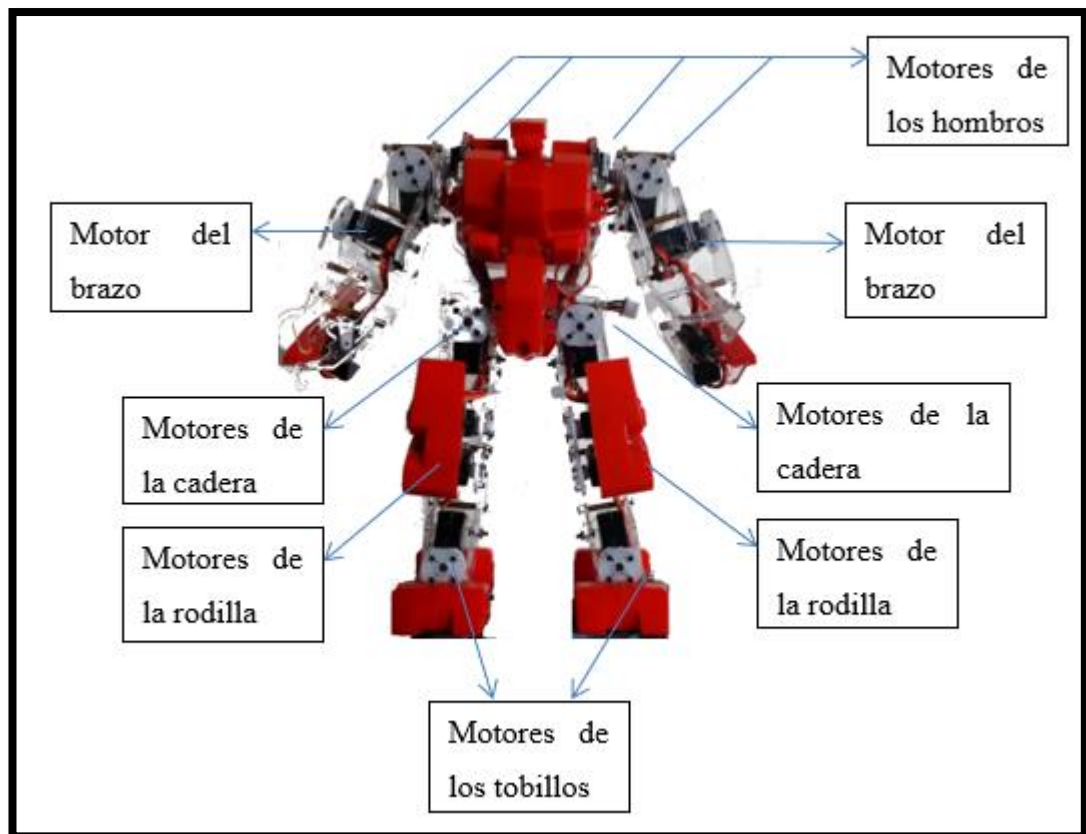


Diagrama de flujo para el funcionamiento del robot, Elaborado por: Edwin Salvador

Actualmente el funcionamiento de los robots humanoides únicamente se lo hace mediante Smart Phones o tablets, para lo cual el piloto del robot debe tener experiencia en el manejo de controles, para solucionar el inconveniente de tener una destreza previa para el control de este tipo de robots. Para esta propuesta se realiza el control mediante un traje, en la Figura 3.1 se muestra el diagrama de flujo para que funcione el robot con el traje.

Para el proceso de control del robot se debe identificar sus partes, así como también donde están ubicados los actuadores, que en el caso del presente proyecto serán servo motores digitales, en la Figura 3.2 se identifican la ubicación de los actuadores.

Figura 3.2 Ubicación de actuadores



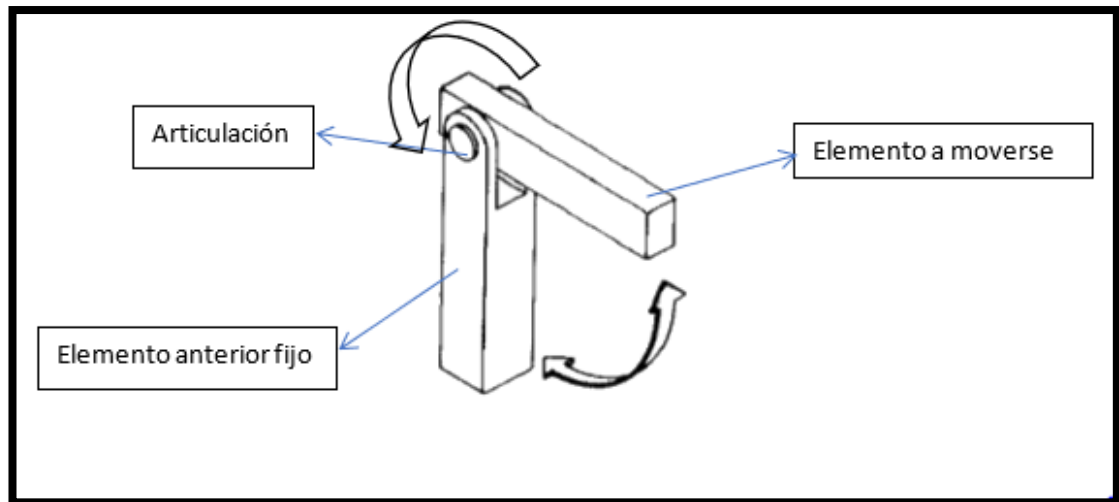
Ubicación de los actuadores en el robot AI.frame, Elaborado por: Edwin Salvador

### 3.2 Cinemática directa

Para realizar la cinemática directa del robot, se considera que cada articulación está anclada a un punto fijo y a partir de ese punto se realiza la lectura de cada sensor que coincide con cada motor ubicado en el robot, al momento de obtener los resultados de las operaciones, se debe tener presente que la información a ser enviada al motor no son ángulos, son posiciones de cada servo digital; para evitar que los servos sufran daños y evitar daños en la estructura del robot, se debe limitar el movimiento de los mismos.

Para determinar los rangos máximos de movimiento, se debe conocer el servo que se moverá, ya que hay articulaciones en las cuales no se puede llegar a una posición dada, esto se produce por la construcción misma del robot, en ciertos casos el servo puede ir desde la posición 500 la 1500 o en otros casos de 1500 a 2500 según la posición del programa Servo Rhythm Controller, dependiendo de que articulación se desee mover. En la Figura 3.3 se observa la composición de una articulación.

Figura 3.3 Articulación



Articulación y sus elementos modelados en Adobe Illustrator, Elaborado por: Edwin Salvador

Al momento de aplicar el concepto de cinemática en el robot, los cálculos que se realizan se reducen a ser uno a uno, lo que quiere decir, una señal de sensor recibida con un elemento a ser movido.

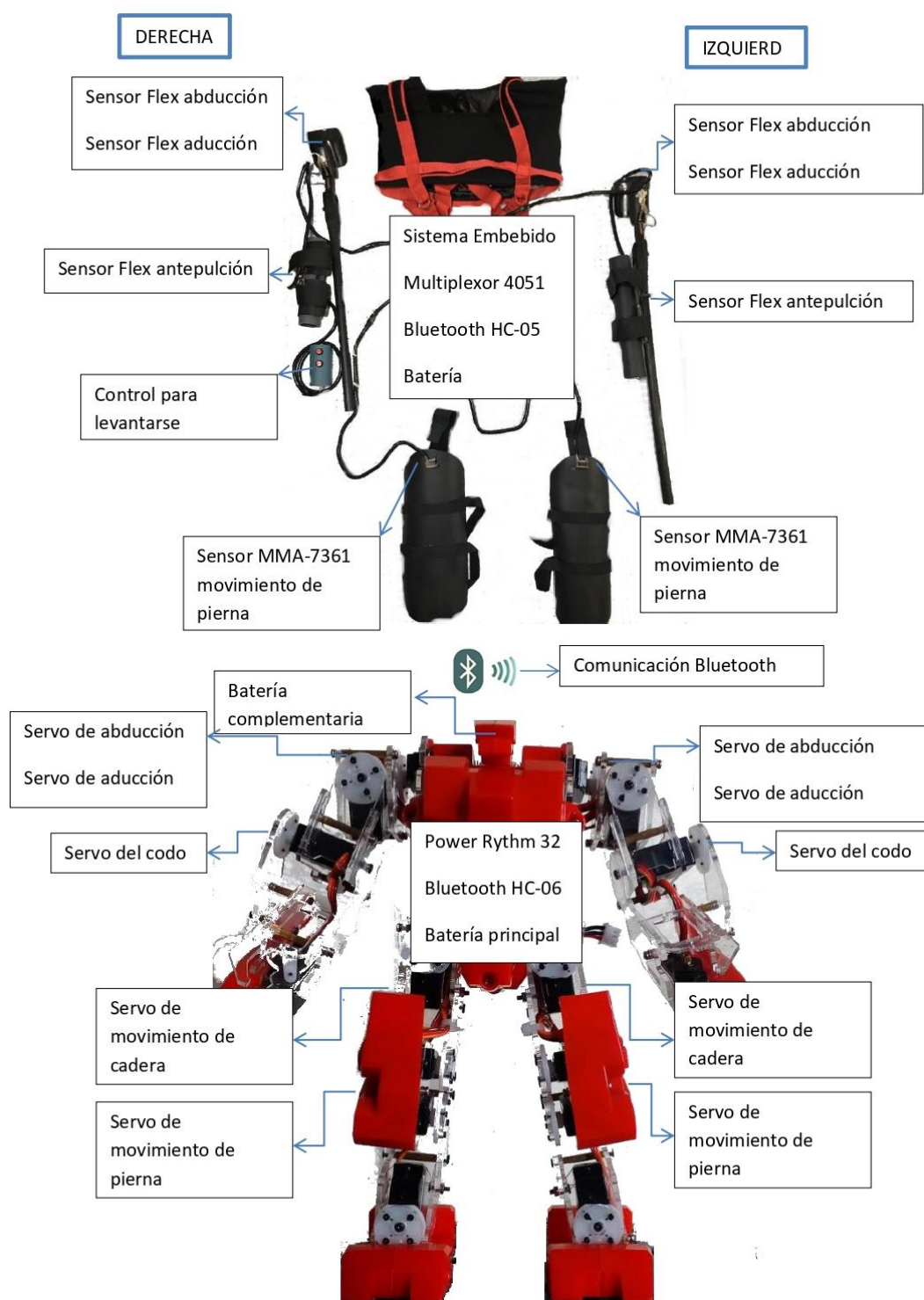
Son 12 señales de los sensores, por lo que se puede rotar 12 articulaciones; pese a ser un robot simétrico, los límites en los cuales se mueven las articulaciones no son los mismos esto se debe a que los actuadores están desfasados  $180^\circ$ .

### 3.3 Arquitectura

Se utilizaron 8 sensores Flex en el proyecto, 4 para detectar los ángulos de rotación de los hombros, 2 para detectar el ángulo de curvatura de los codos y 2 sensores giroscópicos para detectar la posición de las piernas, para cada uno de los sensores se realiza un divisor de tensión independiente, por tal motivo solamente para detectar las señales emitidas por este grupo de sensores se utilizan 8 ADC, para lo cual se implementaron multiplexores análogos, los que servirá para no utilizar todas las DAC del sistema embebido, ya que existen más sensores para ser utilizados, todos estos son sensores análogos.

En la Figura 3.4 se muestra la arquitectura del sistema tanto emisor como receptor, del traje y del robot AI.frame Apollo respectivamente.

Figura 3.4 Arquitectura del Sistema

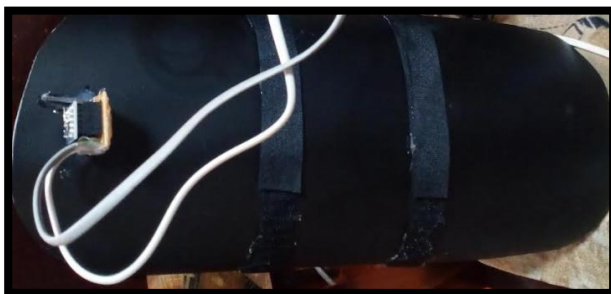


Arquitectura y comunicación del Sistema, Elaborado por: Edwin Salvador

El Arduino DUE se utilizó para recibir las señales de los sensores, procesar la información, acondicionar los valores y enviar la información vía Bluetooth hacia el robot, para que éste imite los movimientos del operador lo más real posible y en un corto tiempo.

Las partes del traje tienden a parecer un exoesqueleto debido como se observa en la Figura 3.5, ya que sus partes son rígidas para montar los sensores y recibir las señales; y a la vez articulado para poder seguir los movimientos del operador.

Figura 3.5 Traje Inferior



Traje inferior con la placa MMA 7361, Elaborado por: Edwin Salvador

La tabla 3.1 representa los valores análogos que el sensor MMA 7361 proporciona a medida que rota en el eje X y en el eje Y, estos valores son convertidos con una resolución de 10 bits.

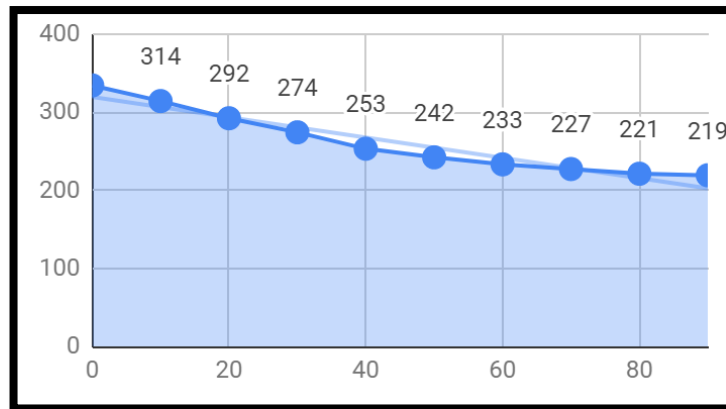
Tabla 3.1 Valores análogos del sensor

<b>Rotación X</b>	<b>Rotación Y</b>	<b>Grados</b>
334	361	0
314	359	10
292	378	20
274	396	30
253	410	40
242	430	50
233	444	60
227	453	70
221	461	80
219	468	90

Valores proporcionados por el sensor MMA 7361, Elaborado por: Edwin Salvador

En la Figura 3.6, se representa la curva de funcionamiento del sensor MMA 7361 específicamente de la rotación en el eje X.

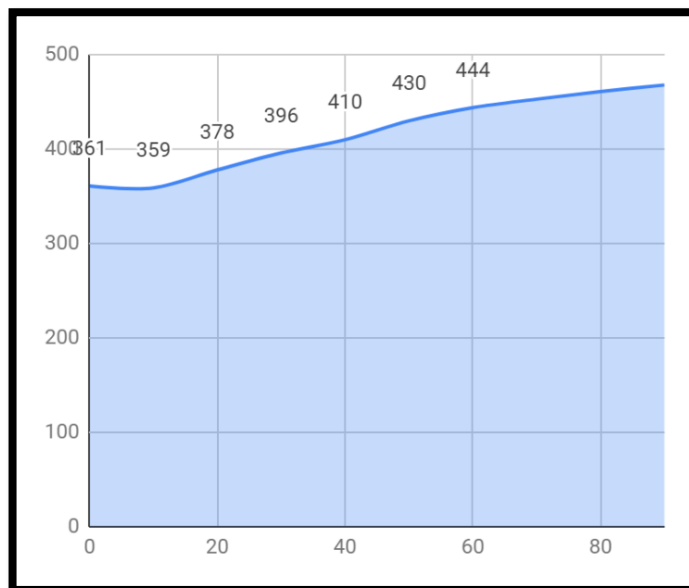
Figura 3.6 Curva eje X



Curva de funcionamiento del sensor en el eje X, Elaborado por: Edwin Salvador

En la Figura 3.7, se representa la curva de funcionamiento del sensor MMA 7361 específicamente de la rotación en el eje Y.

Figura 3.7 Curva eje Y

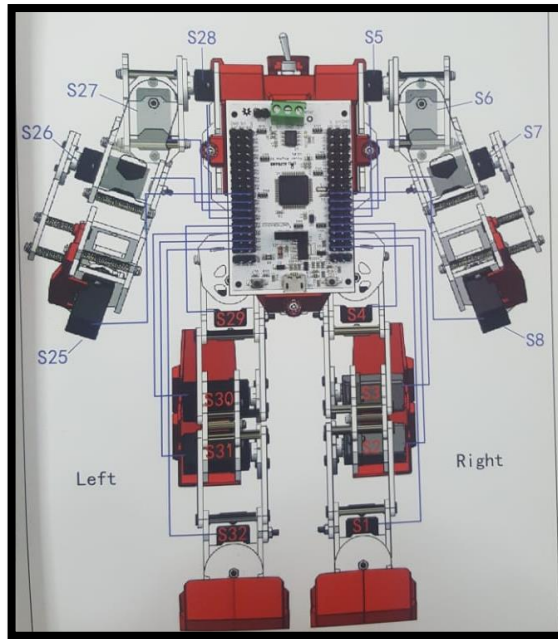


Curva de funcionamiento del sensor en el eje Y, Elaborado por: Edwin Salvador

### 3.4 Acondicionamiento de sensores

En la Figura 3.8 se observa la posición de los motores en el robot, según el fabricante, se utilizará la numeración establecida por el fabricante.

Figura 3.8 Posiciones de motores



Posiciones de los motores en el robot, Fuente: (AI.FRAME)

Las ecuaciones Ec. (3.1), Ec. (3.2), Ec. (3.3), Ec. (3.4), Ec. (3.5), Ec. (3.6), Ec. (3.7), Ec. (3.8), Ec. (3.9), Ec. (3.10); convierten el valor leído de los sensores (x) a un valor para el giro de los actuadores (Y), los sensores están ubicados de tal manera que su valor leído representa los grados de giro de las articulaciones del operador.

$$\begin{aligned}
 &27 \rightarrow Y_0 \text{ Multiplexor 1} \\
 &Y = mx + b \\
 m &= \frac{\text{Valor mínimo del Servo} - \text{Valor máximo del Servo}}{\text{Valor ADC mínimo} - \text{Valor ADC máximo}} \\
 m &= \frac{550 - 1500}{900 - 745} \\
 m &= -6,129 \\
 1500 &= -6,129 \cdot 745 + b \\
 b &= 6066,105 \\
 Y_0 &= -6,129x_0 + 6066,106
 \end{aligned}
 \tag{Ec. (3.1)}$$

$$\begin{aligned}
 &28 \rightarrow Y_1 \text{ Multiplexor 1} \\
 m &= -10,869 \\
 1500 &= -10,869 \cdot 700 + b \\
 b &= 9108,3 \\
 Y_1 &= -10,869x_1 + 9108,3
 \end{aligned}
 \tag{Ec. (3.2)}$$

$$\begin{aligned}
5 &\rightarrow Y_3 \text{ Multiplexor 2} \\
m &= 10,978 \\
1500 &= 10,978 \cdot 700 + b \\
b &= -6184,6 \\
Y_3 &= 10,978x_3 - 6184,6
\end{aligned}
\tag{Ec. (3.3)}$$

$$\begin{aligned}
6 &\rightarrow Y_2 \text{ Multiplexor 2} \\
m &= 7,812 \\
1500 &= 7,812 \cdot 803 + b \\
b &= -4851,15 \\
Y_2 &= 7,812x_2 - 4851,15
\end{aligned}
\tag{Ec. (3.4)}$$

### 3.4.1 Codo Izquierdo

$$\begin{aligned}
26 &\rightarrow Y_5 \text{ Multiplexor 1} \\
180^\circ &\rightarrow 730 \\
90^\circ &\rightarrow 790 \\
m &= -25 \\
2000 &= -25 \cdot 730 + b \\
b &= 20250 \\
Y_5 &= -25x_5 + 20250
\end{aligned}
\tag{Ec. (3.5)}$$

### 3.4.2 Codo Derecho

$$\begin{aligned}
7 &\rightarrow Y_4 \text{ Multiplexor 1} \\
180^\circ &\rightarrow 719 \\
90^\circ &\rightarrow 820 \\
m &= 14,85 \\
1000 &= 14,85 \cdot 718 + b \\
b &= -9677,15 \\
Y_4 &= 14,85x_4 - 9677,15
\end{aligned}
\tag{Ec. (3.6)}$$

### 3.4.3 Pierna Izquierda

$$\begin{aligned}
30 &\rightarrow Y_3 \text{ Multiplexor 2} \\
&\text{"X"} \\
recta &\rightarrow 544 \\
90^\circ &\rightarrow 672 \\
m &= 7,812 \\
1500 &= 7,812 \cdot 544 + b \\
b &= -27,49,72 \\
Y_3 &= 7,812x_3 - 2749,72
\end{aligned}
\tag{Ec. (3.7)}$$



$$\begin{aligned}
29 &\rightarrow Y_4 \text{ Multiplexor 2} \\
&\text{"Y"} \\
&\text{recta} \rightarrow 540 \\
&\text{izquierda} \rightarrow 740 \\
&m = -5 \\
1500 &= -5 \cdot 540 + b \\
b &= 4200 \\
Y_4 &= -5x_4 + 4200
\end{aligned}
\tag{Ec. (3.8)}$$

#### 3.4.4 Pierna Derecha

$$\begin{aligned}
3 &\rightarrow Y_0 \text{ Multiplexor 2} \\
&\text{"X"} \\
&\text{recta} \rightarrow 480 \\
&90^\circ \rightarrow 260 \\
&m = 4,545 \\
500 &= 260 \cdot 4,545 + b \\
b &= -681,81 \\
Y_0 &= 4,545x_0 - 681,81
\end{aligned}
\tag{Ec. (3.9)}$$

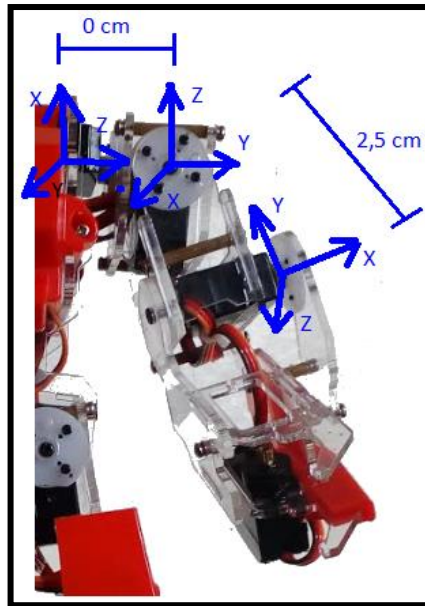
$$\begin{aligned}
4 &\rightarrow Y_1 \text{ Multiplexor 2} \\
&\text{"Y"} \\
&\text{recta} \rightarrow 550 \\
&\text{derecha} \rightarrow 765 \\
&m = 4,651 \\
1500 &= 4,651 \cdot 550 + b \\
b &= -1058,05 \\
Y_1 &= 4,651x_1 - 1058,05
\end{aligned}
\tag{Ec. (3.10)}$$

#### 3.4.5 Cinemática directa del Brazo

La Figura 3.10 representa la posición inicial de la articulación del brazo del robot, en la Figura 3.11 se debió rotar el plano Z en  $90^\circ$ , para la Figura 3.12 se debió rotar el plano X en  $90^\circ$ , de esa manera se consigue establecer la matriz de Denavit-Hartenberg que está representada en la tabla 3.2.

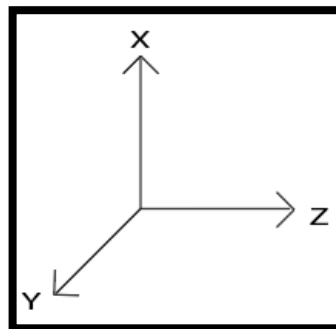
En la figura 3.9 se observa la cinemática del brazo en su posición inicial.

Figura 3.9 Cinemática del brazo



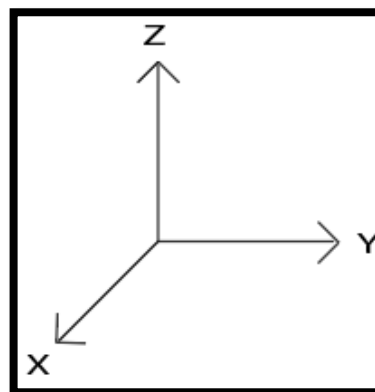
Cinemática inicial del brazo, Elaborado por: Edwin Salvador

Figura 3.10 Posición inicial del brazo



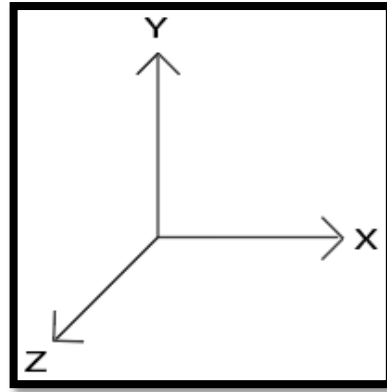
$\theta + 90^\circ$ , Elaborado por: Edwin Salvador

Figura 3.11 Rotación del plano del brazo



$\theta + 90^\circ$ , Elaborado por: Edwin Salvador

Figura 3.12 Posición final del plano del brazo



$\alpha=90^\circ$ , Elaborado por: Edwin Salvador

Tabla 3.2 Parámetros Denavit-Hartenberg (brazo)

#	$\theta$	$d$	$a$	$\alpha$
1	$\theta_1$	0	0	$+90^\circ$
2	$\theta_2+90^\circ$	0	$d$	$+90^\circ$
3	$\theta_3$	0	0	0

Generador de matrices de Denavit-Hartenberg para el brazo, Elaborado por: Edwin Salvador

En la Ec. (3.11), se establecen las matrices para el movimiento de antepulsión de la primera articulación del hombro del robot.

$${}^0A^1 = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(90^\circ) & -\sin(90^\circ) & 0 \\ 0 & \sin(90^\circ) & \cos(90^\circ) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (3.11)}$$

$${}^0A^1 = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) & 0 \\ \sin(\theta) & 0 & -\cos(\theta) & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La Ec. (3.12), se establecen las matrices para los movimientos de aducción y abducción de la segunda articulación del hombro del robot.

$${}_1A^2 = \begin{bmatrix} \cos(\theta_2 + 90^\circ) & -\sin(\theta_2 + 90^\circ) & 0 & 0 \\ \sin(\theta_2 + 90^\circ) & \cos(\theta_2 + 90^\circ) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 2,5 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(90^\circ) & -\sin(90^\circ) & 0 \\ 0 & \sin(90^\circ) & \cos(90^\circ) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}_1A^2 = \begin{bmatrix} -\sin(\theta_2) & 0 & \cos(\theta_2) & -2,5 \sin(\theta_2) \\ \cos(\theta_2) & 0 & \sin(\theta_2) & 2,5 \cos(\theta_2) \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ec. (3.12)

En la Ec. (3.13) se establecen la matriz para el movimiento del codo del robot.

$${}_2A^3 = \begin{bmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 & 0 \\ \sin(\theta_3) & \cos(\theta_3) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (3.13)}$$

En la Ec. (3.14), se visualiza la resolución de las matrices de Denavit-Hartenberg, con las posiciones en los planos X, Y y Z.

$${}_0A^3 = {}_0A^1 \cdot {}_1A^2 \cdot {}_2A^3$$

$${}_0A^3 = \begin{bmatrix} \sin(\theta_1) \sin(\theta_3) - \sin(\theta_2) \cos(\theta_1) \cos(\theta_3) & \sin(\theta_1) \cos(\theta_3) + \sin(\theta_2) \cos(\theta_1) \sin(\theta_3) & \sin(\theta_1) \sin(\theta_2) \cos(\theta_3) - \cos(\theta_1) \sin(\theta_2) \sin(\theta_3) & \sin(\theta_1) \cos(\theta_2) \cos(\theta_3) - \cos(\theta_1) \sin(\theta_2) \sin(\theta_3) \\ -\sin(\theta_2) \sin(\theta_1) \cos(\theta_3) - \cos(\theta_1) \sin(\theta_3) & \sin(\theta_2) \sin(\theta_1) \sin(\theta_3) - \cos(\theta_1) \cos(\theta_3) & \cos(\theta_2) \cos(\theta_3) & -\cos(\theta_2) \sin(\theta_3) \\ 0 & 0 & \cos(\theta_2) \cos(\theta_1) & -2,5 \sin(\theta_2) \cos(\theta_1) \\ \cos(\theta_2) \cos(\theta_1) & -2,5 \sin(\theta_2) \cos(\theta_1) & \cos(\theta_2) \sin(\theta_1) & -2,5 \sin(\theta_2) \sin(\theta_1) \\ \sin(\theta_2) & 2,5 \cos(\theta_2) + 1 & 0 & 1 \end{bmatrix}$$

$$Px = -2,5 \sin(\theta_2) \cos(\theta_1)$$

$$Py = -2,5 \sin(\theta_2) \sin(\theta_1)$$

$$Pz = 2,5 \cos(\theta_2) + 1$$

Ec. (3.14)

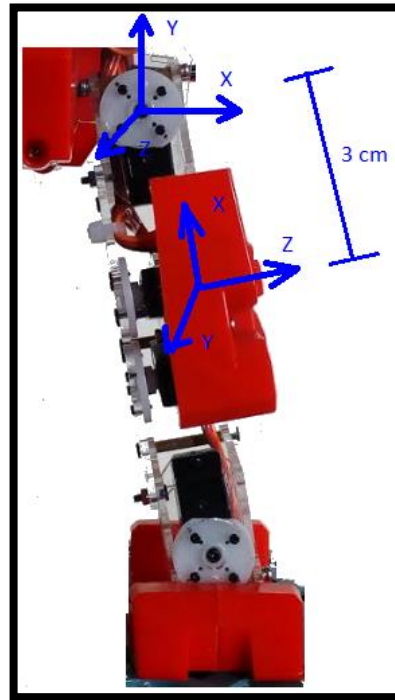
Las ecuaciones Ec. (3.11), Ec. (3.12), Ec. (3.13) y Ec. (3.14), son válidas para el brazo derecho y el brazo izquierdo.

### 3.4.6 Cinemática directa de la Pierna

La Figura 3.14 representa la posición inicial de la articulación de la pierna del robot, en la Figura 3.15 rotó  $90^\circ$  el plano X, de esa manera se consigue establecer la matriz de Denavit-Hartenberg que está representada en la tabla 3.3.

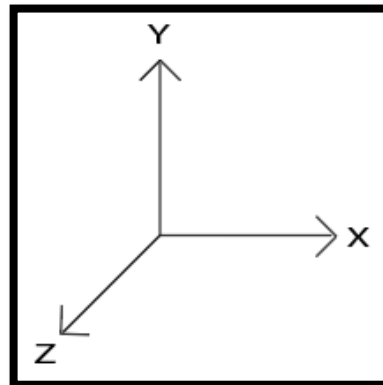
En la figura 3.13 se observa la cinemática de la pierna en su posición inicial.

Figura 3.13 Cinemática de la pierna



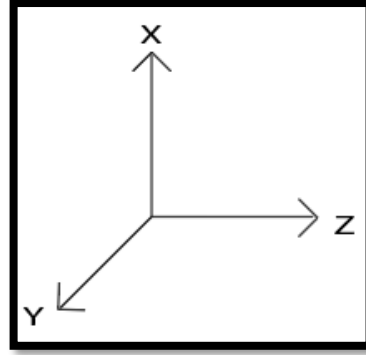
Cinemática inicial de la pierna, Elaborado por: Edwin Salvador

Figura 3.14 Posición inicial de la pierna



Posición inicial del plano para la pierna del robot, Elaborado por: Edwin Salvador

Figura 3.15 Posición final de la pierna



$\alpha=+90^\circ$ , Elaborado por: Edwin Salvador

Tabla 3.3 Parámetros Denavit-Hartenberg (pierna)

#	$\Theta$	D	A	$\alpha$
1	$\theta_1$	0	0	$+90^\circ$
2	$\theta_2+90^\circ$	0	3	0

Generador de matrices de Denavit-Hartenberg para la pierna, Elaborado por: Edwin Salvador

En la Ec. (3.15), se establecen las matrices para el movimiento de rotación de la articulación de la pierna del robot.

$${}^0A^1 = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(90^\circ) & -\sin(90^\circ) & 0 \\ 0 & \sin(90^\circ) & \cos(90^\circ) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (3.15)}$$

$${}^0A^1 = \begin{bmatrix} \cos(\theta_1) & 0 & \sin(\theta_1) & 0 \\ \sin(\theta_1) & 0 & -\cos(\theta_1) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La Ec. (3.16), se establecen las matrices para los movimientos de traslación de la articulación de la pierna del robot.

$${}_1A^2 = \begin{bmatrix} \cos(\theta_2 + 90^\circ) & -\sin(\theta_2 + 90^\circ) & 0 & 0 \\ \sin(\theta_2 + 90^\circ) & \cos(\theta_2 + 90^\circ) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (3.16)}$$

$${}_1A^2 = \begin{bmatrix} \cos(\theta_2 + 90^\circ) & -\sin(\theta_2 + 90^\circ) & 0 & 3 \cos(\theta_2 + 90^\circ) \\ \sin(\theta_2 + 90^\circ) & \cos(\theta_2 + 90^\circ) & 0 & 3 \sin(\theta_2 + 90^\circ) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

En la Ec. (3.17), se visualiza la resolución de las matrices de Denavit-Hartenberg, con las posiciones en los planos X, Y y Z.

$${}_0A^2 = \begin{bmatrix} \cos(\theta_1) \cos(\theta_2 + 90^\circ) & -\cos(\theta_1) \sin(\theta_2 + 90^\circ) \\ \sin(\theta_1) \cos(\theta_2 + 90^\circ) & -\sin(\theta_1) \sin(\theta_2 + 90^\circ) \\ \sin(\theta_2 + 90^\circ) & \cos(\theta_2 + 90^\circ) \\ 0 & 0 \\ \sin(\theta_1) & 3 \cos(\theta_1) \cos(\theta_2 + 90^\circ) \\ -\cos(\theta_1) & 3 \sin(\theta_1) \cos(\theta_2 + 90^\circ) \\ 0 & 3 \sin(\theta_2 + 90^\circ) \\ 0 & 1 \end{bmatrix} \quad \text{Ec. (3.17)}$$

$$Px = -3 \cos(\theta_1) \sin(\theta_2)$$

$$Py = -3 \sin(\theta_1) \sin(\theta_2)$$

$$Pz = -3 \cos(\theta_2)$$

Las ecuaciones Ec. (3.15), Ec. (3.16) y Ec. (3.17), son válidas para la pierna derecha y la pierna izquierda.

### 3.5 Diagrama de flujo

La Figura 3.16, representa el diagrama de flujo de los procesos y subprocesos que realiza el programa para copiar los movimientos del operador en el robot.

Figura 3.16 Flujo del programa

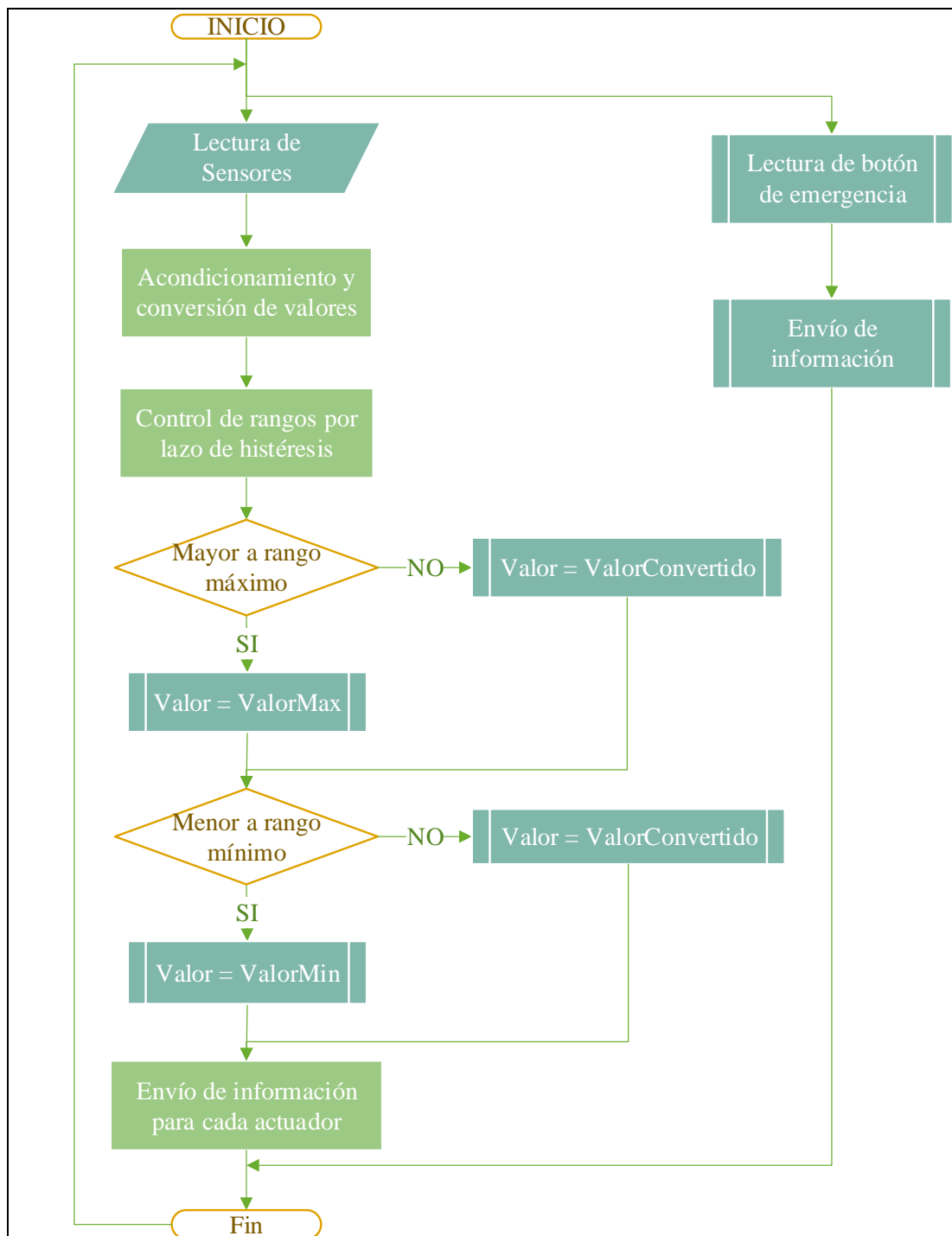


Diagrama de flujo del programa, Elaborado por: Edwin Salvador



### 3.6 Diagrama de bloques

En la Figura 3.17 se muestra el diagrama de conexiones desde los sensores hacia los multiplexores y la interacción de la información (datos) que realizan los multiplexores con el sistema embebido, además la conexión que este tiene con la tarjeta controladora y a la vez las conexiones con los servos motores respectivos.

Figura 3.17 Diagrama de bloques

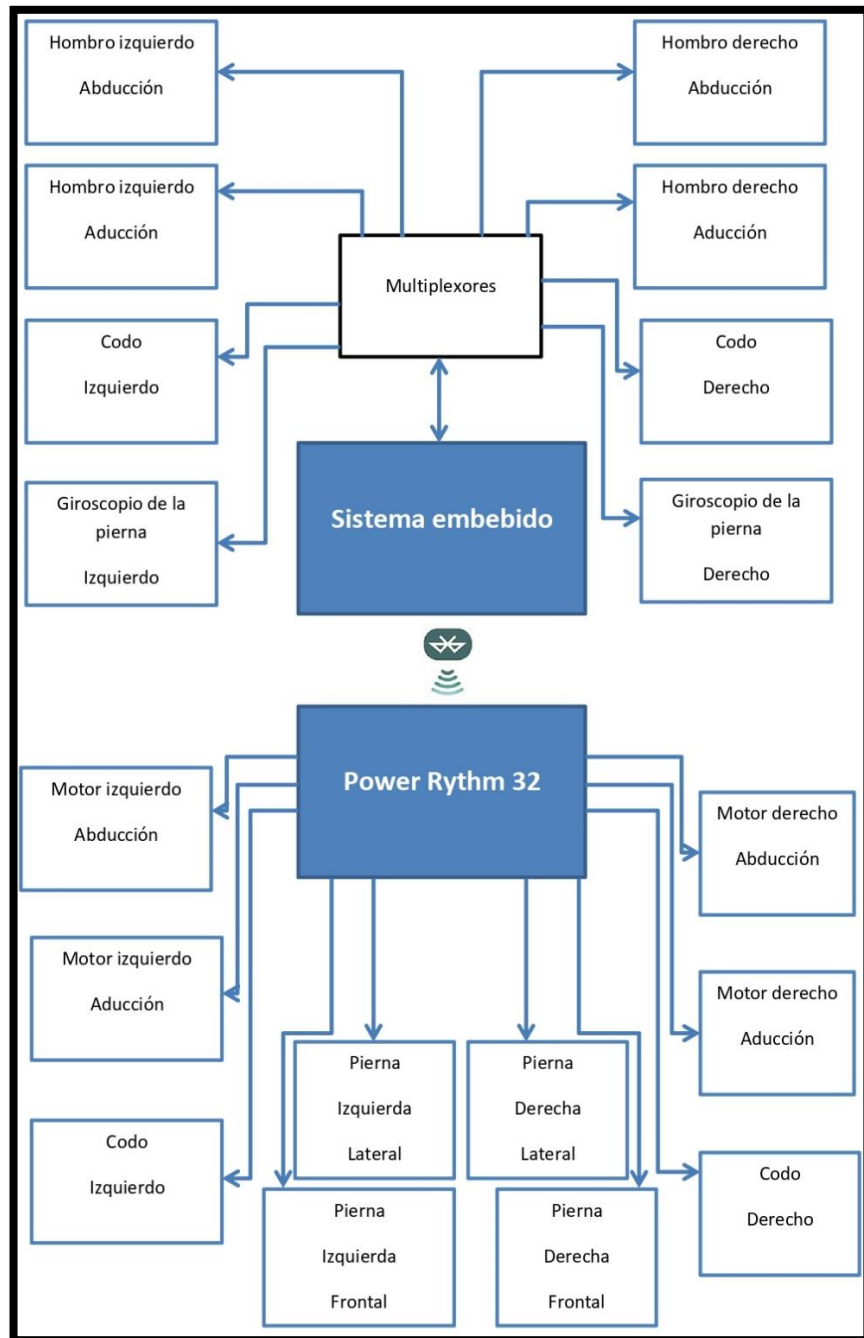


Diagrama de bloques de las conexiones, Elaborado por: Edwin Salvador

### 3.7 Código fuente

En la Figura 3.18 se muestra una selección de salidas digitales, estas salidas al momento de activarse o desactivarse hace que el multiplexor seleccione una entrada para leer un sensor diferente, además se observa un retraso de 10 milisegundos, este retraso sirve para permitir que el multiplexor realice el respectivo switcheo, este retraso es necesario ya que el sistema embebido procesa mucho más rápido que el multiplexor utilizado, y por esa diferencia de velocidades es necesario dar tiempos para que el multiplexor responda de una manera adecuada, si no se dan tiempos de respuesta, podemos tener lecturas erróneas o lecturas de sensores diferentes.

Figura 3.18 Salidas digitales

```
digitalWrite(S0, HIGH);  
digitalWrite(S1, LOW);  
digitalWrite(S2, LOW);  
delay(10);
```

Selección de salidas digitales, Elaborado por: Edwin Salvador

En la Figura 3.19 se muestra la lectura de 2 entradas análogas porque existen dos multiplexores diferentes.

Figura 3.19 Lectura análoga

```
lect[1]=analogRead(1);  
lect1[1]=analogRead(2);
```

Lectura de 2 salidas análogas, Elaborado por: Edwin Salvador

En la Figura 3.20 se muestran condiciones para un control por histéresis que evita que los motores giren más de lo que deben y así evitar los daños ya sea en la estructura del robot o daños en los mismos motores, los daños en los motores se producen cuando se ven forzados a llegar a los límites y permanecer mucho tiempo en una posición.

Figura 3.20 Control por histéresis

```
if(lect[1]<500)
{
    lect[1]=500;
}
if(lect[1]>1500)
{
    lect[1]=1500;
}
```

Condiciones para un control por histéresis, Elaborado por: Edwin Salvador

En la Figura 3.21 se observa que para el envío de datos se establece primero una condición con las variables “a” y “b”, las cuales están destinadas para saber si una interrupción está activa. Las interrupciones en la programación están destinadas para ejecutar comandos establecidos para que el robot se ponga de pie cuando se haya caído, sea de manera frontal o posterior.

Figura 3.21 Condición

```
if(a==0 || b==0)
{
```

Condición de interrupción para que se ponga de pie el robot, Elaborado por: Edwin Salvador

En la Figura 3.22 se muestra el código para el envío propiamente de información, lo que primero se debe hacer es enviar el número del motor a ser activado, para esto se coloca un “#” seguido por el número del motor y la letra “P”, de debe mantener esa secuencia para que la placa controladora de servos lea la información, de no mantener el formato en ese orden, la información es desechada; lo siguiente que se debe enviar es la posición del motor, pero esta información debe ser un número decimal y entero, siempre debe estar parametrizado entre los valores máximos y mínimos establecidos por el fabricante

Figura 3.22 Código de envío

```
Serial2.print("#27P");
Serial2.print(lect[0],DEC);
```

Código de envío de información para activar los motores, Elaborado por: Edwin Salvador

En la Figura 3.23 se muestra la última parte de la secuencia de comandos, este es el tiempo que debe tardar el motor en tomar la posición indicada, el formato para enviar el tiempo debe ser el siguiente: “T” seguido de un entero que represente el tiempo en milisegundos y “!”, esto permitirá que la orden pueda ser leída por la tarjeta controladora.

Figura 3.23 Final de la secuencia de comandos

```
Serial2.println("T50!");
```

Representa el tiempo del motor en tomar una posición, Elaborado por: Edwin Salvador

En la Figura 3.24 se muestran las condiciones que deben ser igual a uno cuando el robot se cae, al momento que el robot es derribado el operador mediante dos pulsadores puede controlar el levantamiento. Cuando el robot es derribado y cae boca arriba, se debe enviar la orden para que “a” sea igual a 1, y, cuando el robot se cae boca abajo, la variable “b” debe ser igual a 1; en ese momento cuando se da una u otra condición una secuencia de comandos sugeridos se envían al robot para que se ponga de pie.

Figura 3.24 Condiciones para levantar al robot

<pre>if(a==1) {</pre>	<pre>if(b==1) {</pre>
-----------------------	-----------------------

Condiciones para levantar el robot de frente o posterior, Elaborado por: Edwin Salvador

En la Figura 3.25 se muestra la creación de las interrupciones externas y el cambio de estado de las variables que se utilizan para que el robot se ponga de pie.

Figura 3.25 Interrupciones y cambio de estados

```
void arriba()
{
  a=1;
}

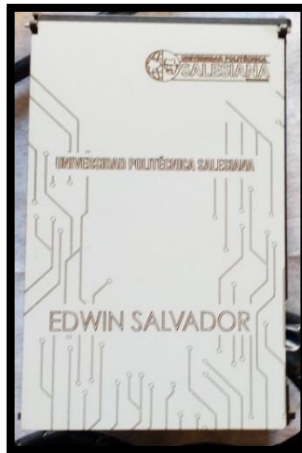
void abajo()
{
  b=1;
}
```

Interrupciones y cambio de estado para levantar el robot, Elaborado por: Edwin Salvador

### 3.8 Diseño del traje

En la Figura 3.26 se muestra la caja dónde va contenido el sistema embebido, los multiplexores, el módulo bluetooth y la batería; esta caja está hecha en madera MDF, recortada y gravada con láser, cuenta con cortes en su estructura para el ingreso de las señales de los sensores provenientes del traje.

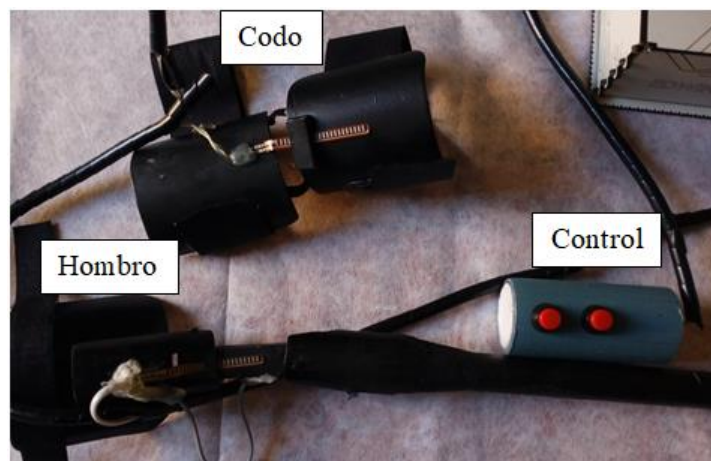
Figura 3.26 Caja contenedora



Caja grabada porta placas, Elaborado por: Edwin Salvador

En la Figura 3.27 se observa las partes del traje fabricadas en materiales de madera y PVC dichas partes forman el brazo derecho, el cual está conformado por el hombro, el codo y dos pulsadores, los mismos que cumplen con la funcionalidad de poner de pie al robot.

Figura 3.27 Traje brazo derecho



Traje brazo derecho (hombro, codo, control lado derecho), Elaborado por: Edwin Salvador

La Figura 3.28 muestra las partes del hombro derecho del traje, el cual posee 2 grados de libertad y está formado por 2 sensores Flex, el material en cual está fabricado es madera y PVC, estos materiales ofrecen la robustez y maleabilidad necesaria para dar la forma deseada.

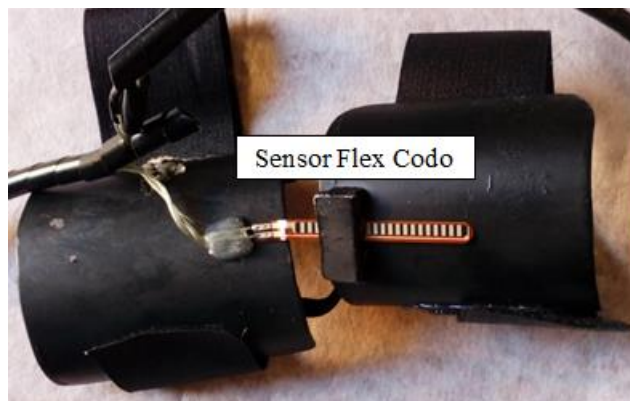
Figura 3.28 Hombro derecho



Hombro derecho del traje, Elaborado por: Edwin Salvador

En la Figura 3.29 se observa el codo del traje, el cual está hecho en PVC por su rigidez, posee 1 grado de libertad, esta estructura está sujeta con elástico para proporcionar la flexibilidad que se necesita para que el operador pueda moverse con cierta libertad y poder llevar a cabo las técnicas en el combate.

Figura 3.29 Codo derecho



Codo derecho del traje, Elaborado por: Edwin Salvador

En la Figura 3.30 se muestra el control que está fabricado en PVC, este control se lo sujeta con la mano derecha; los pulsadores son plásticos y sirven para cuando el robot sea derribado ya sea de frente o posterior, estos botones envían una señal al sistema

embebido, el cual ejecuta una acción de paro a los movimientos para enviar los comandos que harán que el robot se ponga de pie.

Figura 3.30 Control de caídas



Control de caídas, Elaborado por: Edwin Salvador

En la Figura 3.31 se muestra el hombro izquierdo y el codo izquierdo del traje que cumple con las mismas características que el hombro y codo derecho, su diferencia es que en este lado del traje no existen controles extra como los pulsadores del lado derecho del traje.

Figura 3.31 Hombro y codo izquierdo



Hombro y codo izquierdo del traje, Elaborado por: Edwin Salvador

En la Figura 3.32 se observan las musleras, las cuales tienen un sensor giroscópico para detectar los movimientos frontales, posteriores y laterales, están elaboradas con material PVC y poseen sujetadores elaborados con velcro para definir la posición ideal según las características físicas del operador.

Figura 3.32 Musleras



Musleras del traje, Elaborado por: Edwin Salvador

En la Figura 3.33 se observa el traje según las posiciones estables de izquierda y derecha de las partes que lo conforman, todos los cables que van hacia los sensores están cubiertos por un material práctico llamado espiral, con esto se puede conservar el orden y proteger los cables para evitar que se enreden o en el peor de los casos que se puedan romper.

Figura 3.33 Traje completo



Traje completo, Elaborado por: Edwin Salvador



En la Figura 3.34 se pueden identificar dos partes: la primera parte es una hombrera con espaldar que brinda el soporte y la fijeza necesaria para apoyar a lo que vendría a ser el hombro del traje en el que están montados los sensores, sin este soporte los hombros del traje se tienden a deslizar y los valores obtenidos por los sensores son erróneos; la segunda parte es un arnés el cual brinda la sujeción necesaria para que las hombreras con espaldar se mantenga en su lugar, también posee líneas de velcro para adherir los hombros en el arnés; además en la parte posterior del arnés se encuentra un bolsillo en el cual se guarda la caja que contiene el sistema embebido, de esta manera el arnés brinda comodidad y el operador únicamente se preocupa por la pelea que se lleva a cabo ya que el traje se sostiene de una manera efectiva sin riesgos de caerse.

Figura 3.34 Arnés y hombrera con espaldar



Arnés y hombrera con espaldar, Elaborado por: Edwin Salvador

## CAPÍTULO 4

### PRUEBAS Y RESULTADOS

#### 4.1 Pruebas

En el presente capítulo se observará los movimientos realizados por el operador e imitados por el robot, los movimientos que se realizarán son golpes y desplazamientos que se necesitan para un combate.

En la Figura 4.1 se observa el movimiento articular de los hombros, esto se debe a que los golpes realizados son rectos, un golpe recto es aquel en el que se estira el brazo en su totalidad sin flexionar el codo. Los golpes rectos son considerados como golpes básicos y elementales en un combate, cuando se realizan estos golpes un brazo se estira mientras el otro permanece recogido para cubrir o alistar el siguiente golpe.

Figura 4.1 Golpes Rectos



Golpes rectos izquierdo y derecho, Elaborado por: Edwin Salvador

En la Figura 4.2 se realizan movimientos combinados, en los cuales se estiran los brazos de forma lateral y se flexionan el codo de una manera sincronizada. Este tipo de movimientos permite verificar la sincronización entre el operador y el robot.

Figura 4.2 Movimiento combinado



Movimiento combinado hacia la derecha e izquierda, Elaborado por: Edwin Salvador

En la Figura 4.3 se realizan movimientos de abducción con los codos cerrados a 90°.

Figura 4.3 Movimiento de abducción



Abducción con codos a 90°, Elaborado por: Edwin Salvador

En la Figura 4.4 se realizan movimientos de antepulsión con los codos cerrados a 90°.

Figura 4.4 Movimientos de Antepulsión



Antepulsión subiendo y bajando los brazos con codos a 90°, Elaborado por: Edwin Salvador

En la Figura 4.5 se realizan pruebas de movimiento de las piernas en forma lateral para lo cual se observa que el robot realiza el mismo movimiento de una manera sincronizada. Los movimientos laterales permiten el desplazamiento del operador y del robot de izquierda a derecha y viceversa.

Figura 4.5 Movimiento lateral de piernas



Movimiento lateral de piernas hacia la derecha e izquierda, Elaborado por: Edwin Salvador

En la Figura 4.6 se realiza movimientos frontales, cuando el operador realiza estos movimientos el robot debe agacharse para conservar el equilibrio y su verticalidad.



Figura 4.6 Movimientos frontales de piernas



Movimientos frontales de la pierna izquierda y derecha, Elaborado por: Edwin Salvador

En la Figura 4.7 se realiza un doble golpe de puños rectos.

Figura 4.7 Golpe doble



Golpe doble de puño cerrado, Elaborado por: Edwin Salvador

## 4.2 Resultados

En la tabla 4.1 se muestran los tiempos aproximados que tiene de respuesta el robot respecto a los movimientos del operador, estos valores al ser medidos externamente se los debe promediar para tener un resultado aproximado.

Tabla 4.1 Tiempo de respuesta

	Motor izquierdo Abducción (ms)	Motor izquierdo Aducción (ms)	Codo Izquierdo (ms)	Pierna Izquierda lateral(ms)	Pierna Izquierda frontal (ms)	Pierna Derecha lateral (ms)	Pierna Derecha frontal (ms)	Codo Derecha (ms)	Motor derecho Abducción (ms)	Motor derecho Aducción (ms)
1	111	109	109	111	108	108	113	111	113	113
2	113	107	113	113	109	109	107	113	109	110
3	110	113	112	110	113	113	109	108	111	112
4	114	108	112	112	112	112	110	108	113	108
5	108	112	108	108	110	112	111	112	108	113
	111,2	109,8	110,8	110,8	110,4	110,8	110	110,4	110,8	111,2

Tiempos aproximados de respuesta del robot, Elaborado por: Edwin Salvador

## **CAPÍTULO 5**

### **CONCLUSIONES**

Se diseñó un traje de estructura rígida y maleable, utilizando materiales como madera, PVC y textiles, el traje soportará los sensores y brindará comodidad al operador, los sensores son los encargados de captar los movimientos realizados por el operador los cuales serán codificados y enviados al robot humanoide.

En la investigación realizada sobre los sensores se dedujo que los más indicados para la realización del proyecto son los sensores de salida análoga los cuales se clasifican en dos grupos, el primero son sensores resistivos y el segundo son sensores giroscópicos, este tipo de sensores ofrecen un tiempo de respuesta corto y la adquisición de los datos es rápida.

Los cálculos realizados con cinemática directa permitieron establecer las posiciones de las diferentes articulaciones, las posiciones son determinadas por los movimientos realizados por el operador del robot.

Para procesar la información adquirida de los sensores implantados se ha utilizado el sistema embebido Arduino DUE, el sistema embebido seleccionado procesa la información a una velocidad de 84 MHz, procesamiento y envío de la información de los sensores.

El traje que se utilizó para la realización del proyecto consta de tres partes: hombreras que fueron diseñadas de un material rígido para que mantenga los sensores de los hombros alineados, el arnés que servirá para sujetar el exoesqueleto que fue diseñado para simular las extremidades con sus respectivas articulaciones, en las articulaciones se encuentran colocados los sensores que enviarán la información de los movimientos a una caja receptora que se encuentra sujeta en el arnés.

La comunicación del sistema embebido es realizada vía Bluetooth, esto se debe a que la tarjeta controladora de servos (rithim 32) tiene comunicación serial.

Se debe mantener las articulaciones del traje alineadas con las articulaciones del operador para que las lecturas de los sensores sean lo más precisas posibles, además se debe sujetar de manera correcta para que las mismas se mantengan en su posición cuando el operador realice los movimientos, la espalda del operador no debe



sobrepasar los 49cm de ancho ya que, si llegase a sobrepasar, las lecturas de los sensores serian erróneas.

Para que el robot pueda ser homologado e incluso para el desarrollo de la pelea se realizó una programación extra para que el robot se ponga de pie al momento de ser derribado o sufra un tropiezo, esta programación extra se ejecuta utilizando pulsadores ubicados en un control manual.

La calibración de los sensores se la realizó en los ángulos de  $90^\circ$  y  $180^\circ$ , para que de esa manera las personas puedan utilizar el traje y los valores de los sensores no sean mayormente afectados, y lograr así que el desempeño del robot no vea perjudicado.

Según las pruebas realizadas y los tiempos de ejecución de los movimientos adquiridos, se pudo concluir que los movimientos son en tiempo real, esto se puede estimar ya que al observar en la tabla 4.1 los valores estos son muy bajos y la ejecución de los movimientos es muy rápida.

## **CAPÍTULO 6**

### **RECOMENDACIONES**

En caso de necesitar reemplazar una pieza del robot se recomienda utilizar piezas originales, ya que el reemplazo inadecuado de sus partes puede provocar la ruptura de la estructura del robot.

Se debe mantener las baterías del robot y del traje completamente cargadas antes de su uso, ya que la carga inadecuada de las baterías provocará lecturas erróneas.

Una vez configurados los módulos Bluetooth, no deben ser manipulados ya que la configuración está programada para que los mismos se conecten automáticamente al momento del encendido.

Se debe utilizar el robot sobre superficies no ásperas para que el robot tenga la libertad de movimientos necesarios y no sea limitado por el rozamiento entre la superficie y el robot.

## REFERENCIAS

- Acero Jáuregui, J. (22 de Abril de 2013). *Grados de Libertad en el Movimiento Humano*. Obtenido de Grupo Sobre Entrenamiento (G-SE): <https://g-se.com/grados-de-libertad-en-el-movimiento-humano-bp-757cfb26d3868e>
- AI.Frame. (6 de Mayo de 2016). *Servocontrolador de 32 canales Power Rhythm 32*. Obtenido de Bluecomtech: <http://web.bluecomtech.com/AI.Frame/aiframe.me/product/10507a0.html?language=en>
- AI.FRAME. (s.f.). *APOLLO ASSEMBLY MANUAL*.
- Arduino. (2019). *Productos Arduino*. Obtenido de Arduino: <https://www.arduino.cc/en/Main/Products>
- Bricogeek. (19 de Diciembre de 2015). *Arduino original -> Arduino Due*. Obtenido de BricoGeek: <https://tienda.bricogeek.com/arduino-original/476-arduino-due.html>
- Campos Torres, G. (2016). *Centro de Datos: ¿almacenamiento de computadoras?* Obtenido de SemanticWebBuilder: [http://www.semanticwebbuilder.org.mx/es\\_mx/swb/Sistemas\\_Embebidos\\_Innovando\\_hacia\\_los\\_Sistemas\\_Inteligentes](http://www.semanticwebbuilder.org.mx/es_mx/swb/Sistemas_Embebidos_Innovando_hacia_los_Sistemas_Inteligentes)
- Castillo Rodríguez, F. (20 de Mayo de 2012). *Biomecánica del hombro*. Obtenido de Slideshare: <https://es.slideshare.net/fisioterapeuta54/biomecanica-del-hombro-13003613>
- Components10. (S/F). *HC-05 - Bluetooth Module*. Obtenido de COMPONENTS10: <https://components101.com/wireless/hc-05-bluetooth-module>
- Computer Hoy. (2 de Octubre de 2018). *Este robot humanoide podría reemplazar la mano de obra humana*. Obtenido de TN8: <https://www.tn8.tv/tecnologia/458511-este-robot-humanoide-podria-reemplazar-la-mano-de-obra-humana/>

Curiosoando.com. (24 de Abril de 2014). *¿Qué es una pila o batería seca?* Obtenido de Curiosoando: <https://curiosoando.com/que-es-una-pila-o-bateria-seca>

Departamento de Ingeniería Eléctrica, Electrónica y de Control. (2017). *Controladores industriales de diseño de alto nivel*. Obtenido de UNED, Ingeniería Eléctrica, Electrónica y de Control: [http://www.ieec.uned.es/investigacion/Dipseil/PAC/archivos/Informacion\\_de\\_referencia\\_ISE5\\_3\\_1.pdf](http://www.ieec.uned.es/investigacion/Dipseil/PAC/archivos/Informacion_de_referencia_ISE5_3_1.pdf)

Electronicos Caldas. (S/F). *482 Modulo Bluetooth HC-06*. Obtenido de Electronicos Caldas: <https://www.electronicoscaldas.com/es/modulos-rf/482-modulo-bluetooth-hc-06.html>

Graynomad. (8 de Junio de 2013). *Arduino Due > Due pinout diagram*. Obtenido de Arduino: <https://forum.arduino.cc/index.php?topic=132130.0>

Grupo de Robots Argentina. (5 de Noviembre de 2018). *Servos: características básicas*. Obtenido de Robots Didácticos: <http://robots-argentina.com.ar/didactica/servos-caracteristicas-basicas/>

Haro, J. (20 de Mayo de 2015). *MMA7361 Sensor Acelerometro*. Obtenido de HETPRO, Arduino, Sensores Analógicos: <https://hetpro-store.com/TUTORIALES/mma7361-sensor-acelerometro/>

Kopiev, G. (21 de Noviembre de 2017). RPP NOTICIAS. *Toyota presentó un robot humanoide controlado mediante exoesqueleto*, pág. 1.

MIT MOVILTRONICS Ltda. (s.f.). *Módulo Acelerómetro 3 Ejes Analogo (MMA7361)*. Obtenido de MIT Moviltronics, Módulos, Sensores: <https://moviltronics.com.co/sensores/92-modulo-acelerometro-mma7361.html>

Oñate, I. (24 de Octubre de 2012). *Entorno de desarrollo para diseños embebidos*. Obtenido de Industria Embebida Hoy: <https://www.industriaembeidahoy.com/entorno-de-desarrollo-para-disenos-embebidos/>

Rambal. (12 de Septiembre de 2017). *Sensores*. Obtenido de Rambal Automatización y Robótica: <http://rambal.com/presion-peso-nivel-flex/250-sensor-flex.html>

Ramírez Benavides, K. (s.f.). *Cinemática Directa del Robot*. Obtenido de Universidad de Costa Rica, Robótica: <http://www.kramirez.net/Robotica/Material/Presentaciones/CinematicaDirectaRobot.pdf>

RobotHumanoide.Top. (2018). *LOS MEJORES ROBOTS HUMANOIDES ESTÁN AQUÍ*. Obtenido de RobotHumanoide.Top: <https://robothumanoide.top/>

Sensor Products & INC. (2019). *The Benefits of Using Flex Sensors*. Obtenido de Sensorprod.com, Precision Bend Sensor, TactilusFlex: [https://www.sensorprod.com/campaign/bend-sensor-white-paper/index.php?k=Flex&mcode=GC-Flex%20-%20Flex%20Sensor-02&gclid=CjwKCAjw8-LnBRAyEiwA6eUMGs16uLShuMcx\\_wx6P9xZgJ9ngvgA5Tau3W5QevoVCWFetsFqx48UvxoCvK8QAvD\\_BwE](https://www.sensorprod.com/campaign/bend-sensor-white-paper/index.php?k=Flex&mcode=GC-Flex%20-%20Flex%20Sensor-02&gclid=CjwKCAjw8-LnBRAyEiwA6eUMGs16uLShuMcx_wx6P9xZgJ9ngvgA5Tau3W5QevoVCWFetsFqx48UvxoCvK8QAvD_BwE)

*Sistemas Embebidos: Innovando hacia los Sistemas Inteligentes*. (2016). Obtenido de SemanticWebBuilder: [http://www.semanticwebbuilder.org.mx/es\\_mx/swb/Sistemas\\_Embebidos\\_Innovando\\_hacia\\_los\\_Sistemas\\_Inteligentes\\_](http://www.semanticwebbuilder.org.mx/es_mx/swb/Sistemas_Embebidos_Innovando_hacia_los_Sistemas_Inteligentes_)

SM. (10 de Enero de 2017). *Empezando con el Arduino Due*. Obtenido de Arduino: <https://www.arduino.cc/en/Guide/ArduinoDue>

Texas Instruments Incorporated. (2017). *CD405xB CMOS Single 8-Channel Analog Multiplexer/Demultiplexer With Logic-Level Conversion*. Obtenido de TEXAS INSTRUMENTS: <http://www.ti.com/lit/ds/symlink/cd4051b.pdf>

Torrejón, V. M. (2011). *ANÁLISIS DE UN ALGORITMO DE CONTROL PARA UN. MADRID: UNIVERSIDAD CARLOS III DE MADRID.*

UNISONIC TECHNOLOGIES CO., LTD. (2005). *8-CHANNEL ANALOG MULTIPLEXERS/DEMUTIPLEXERS*. Obtenido de UNISONIC TECHNOLOGIES CO., LTD: <http://pdf1.alldatasheet.com/datasheet-pdf/view/173652/UTC/4051.html>

Universidad de Santiago de Chile. (2018). *Herramientas Matemáticas*. Obtenido de UdeSantiagoVirtual:

<http://www.udesantiagoovirtual.cl/moodle2/mod/book/view.php?id=24916>

ZTONGSWEET Store. (S/F). *Ai.Frame Apollo AIF - 44 - 0 Entry Level Humanoid Robot Kit APP Control Open-source - ROJO CON NEGRO*. Obtenido de Gearbest: <https://es.gearbest.com/store/6444155381973086208.html>